IL MENSILE JACKSON PER GLI UTENTI DI AMIGA

- Il favoloso Amiga 3000
- Il nuovo Workbench 2.0
- Calligrapher
- 3-Demon
- Costruiamoci un videogioco
- Programma Editor/Librarian per MT-32

Sped. in abbonamento postale Gruppo III/70 Taxe perçue (Tassa riscossa) Milano Ferr. Corr.

AMIGA MAGAZINE
ANNO 3 - NUMERO 14
LUGLIO/AGOSTO 1990

Sommario

In copertina il nuovo Amiga 3000

Editoriale

Vacanze: mare, sole e... Amiga

Le novità di questi ultimi mesi sono molto esaltanti: dopo varie voci e mormorii è stato presentato il favoloso Amiga 3000 e di conseguenza la nuova release del Workbench, cioè la 2.0.

Naturalmente, trattandosi di presentazioni eccezionali, abbiamo dato largo spazio della rivista a queste due "chicche" Commodore (con la promessa di seguire in futuro, passo dopo passo, le eccezionali capacità di questi due ottimi prodotti).

Ma Amiga Magazine non è solo ultime novità, questo mese lo speciale è dedicato ad una cosa che Amiga fa meglio di tutti (è o non è Amiga pura immaginazione ?) e cioè la Grafica.

Inoltre, in questo numero parliamo di: 3-Demon, un programma per la creazione di oggetti in tre dimensioni; Calligrapher, un disegnatore professionale di caratteri, di MIDI con un articolo dedicato all'Editor/Librarian dell'MT-32 della arcinota Dr.T's, e... non dimentichiamoci del disco allegato e delle pagine di Transactor.

Infine, abbiamo seguito per tutti i nostri meravigliosi lettori (questo possiamo affermarlo dalle numerose lettere e telefonate che riceviamo in redazione ogni giorno), la manifestazione Bit.Movie e la fiera-mercato ABACUS.

Naturalmente non perdetevi le nostre classiche rubriche (in questo numero abbiamo "allargato" di due pagine lo spazio giochi, visto che le vacanze possono stuzzicare voglie di joystick !).

Per questo mese è tutto, concludiamo augurandovi buone vacanze e arrivederci a Settembre in edicola con un nuovo ed entusiasmante numero di Amiga Magazine.

La Redazione

Direttore Responsabile: Paolo Reina
Coordinamento Tecnico e Redazionale: Massimiliano Anticoli
Tel 02/6948287
Redazione: Stefano Albarelli (On Disk), Romano Tenca (Transactor)
Segreteria di redazione: Elena Ferre - Tel 02/6948254
Art Director: Marcello Longhini
Copertina, Grafica e Impaginazione elettronica: Cristina Turra
Collaboratori: Lucio Bragagnolo, Cesare Palmieri, Liliana Simonetti, Stefano Riva, Sergio Ruocco, Sebastiano Vigna, Daniele Cassanelli, Mauro Ballocchi, Aldo e Andrea Laus, Gianni Biagini Renato Acciardi, Stefano Paganini

Group Publisher: Pierantonio Palerma
Direzione Coordinamento Operativo: Graziella Falaguasta
Publisher Area Consumer: Filippo Canavese
Pubblicità: Filippo Canavese
Direzione Sviluppo Pubblicità: Walter Bussolera

SEDE LEGALE
Via P Mascagni 14 - 20122 Milano

DIREZIONE - REDAZIONE
Via Pola, 9 - 20124 Milano - Tel 02/69481
Fax 02/6948238 Telex 316213 REINA I

PUBBLICITA'
Via Pola, 9 - 20124 Milano - Tel 02/6948254
ROMA - LAZIO E CENTRO SUD Via Lago di Tana, 16
00199 Roma
Tel 06/8380547 - Fax 06/8380637

INTERNATIONAL MARKETING
Tel 02/6948233

DIREZIONE AMMINISTRATIVA
Via Rosellini 12 - 20124 Milano Tel 02/69481
Fax 02/6948238

UFFICIO ABBONAMENTI
Via Rosellini, 12 - 20124 Milano - Fax 02/6948489
Telex 333436GEJ IT - Tel 02/6948490 (nei giorni di martedì, mercoledì, giovedì 14 30 - 17 30)

Prezzo della rivista L 14 000 prezzo arretrato L 28 000

I versamenti vanno indirizzati a
Gruppo Editoriale Jackson SpA
Via Rosellini, 12 - 20124 Milano, mediante l'emissione di assegno bancario o per contanti L'abbonamento puo essere sottoscritto anche utilizzando il c/c postale 11666203

CONSOCIATE ESTERE
GEJ Publishing Group Inc Los Altos Hills
27910 Roble Blanco
94022 California - Tel (001-415-9492028)
Grupo Editorial Jackson - Calle Alcantara, 57
280016 Madrid - Tel 1/4017365

Stampa: F B M (Gorgonzola)
Fotolito: Foligraph (Milano)
Distribuzione: Sodip - Via Zuretti, 25 - 20125 Milano

Il Gruppo Editoriale Jackson e iscritto al Registro Nazionale della stampa al N 117 Vol 2 foglio 129 in data 17/8/1982
Spedizione in abbonamento postale gruppo III/70
Aut Trib di Milano n 102 del 22/2/1988

Associato al

Mensile associato all'USPI
Unione Stampa Periodica Italiana

Testata aderente al C S S T non soggetta a certificazione obbligatoria per la presenza pubblicitaria inferiore al 10%

Il Gruppo Editoriale Jackson pubblica anche le seguenti riviste:

Amiga Magazine Games - Supercommodore 64 e 128
Guida Videogiochi - Pc Software - Fare Elettronica
Bit - Computer Grafica & Desktop Publishing
Informatica Oggi - Informatica Oggi Settimanale
Pc Floppy - Pc Magazine - Automazione Oggi
Trasmissioni Dati e Telecomunicazioni - Elettronica Oggi
EO News settimanale - Strumenti Musicali - Watt
Meccanica Oggi - Strumentazione e Misure Oggi

Link e altro

Spett.le Redazione,
innanzitutto vorrei porgervi le mie congratulazioni per la vostra rivista. Possiedo Amiga da 3 mesi e non ho riscontrato enormi difficoltà, visto che so già programmare in assembler con il Commodore 64.
Per sfruttare al massimo le possibilità di Amiga, dopo un mese di duro lavoro per capire la logica che utilizza questo magnifico computer, ho comprato quattro libri della collana Jackson: AMIGA Basic, AMIGA DOS, AMIGA linguaggio C, AMIGA Assembler.
Ovviamente come tutti sanno per il C e l'assembler è necessario l'utilizzo di programmi speciali (compilatori e assemblatori).
Per il C uso il Lattice C 5.0, ma ho dei problemi quando devo compilare dei programmi che utilizzano la "math. H" in fase di BLINK cioè:

undefined symbols
_sqrt
_cos
ecc.

Ho provato più volte questa fase e non ho mai avuto risultati soddisfacenti; come devo fare ?
Ho saputo che esistono diversi compilatori/assemblatori e il migliore sembrerebbe il DEVPAC della HiSoft, dove lo posso trovare ?

Sampietro Lorenzo
Maspero Claudio

Ti ringraziamo per i complimenti alla rivista, e veniamo subito alle tue domande. Il problema che ti si presenta durante il "linkaggio" avviene perché il BLink non trova le funzioni sqrt e cos.
Per ovviare a questo inconveniente dovresti utilizzare nell'operazione di link anche la libreria matematica lcm.lib in questo modo (è importante che le librerie, situate nell'esempio in LIB:, siano specificate nello stesso ordine con cui sono riportate di seguito):

```
BLink <file>.o TO <file> LIBRARY
LIB:lcm.lib+LIB:lc.lib+LIB:amiga.lib
```

Per quanto riguarda il DEVPAC della HiSoft è distribuito ufficialmente da:

Leader Distribuzione
Via Mazzini, 15 - Casciago (VA)
Tel. 0332/212255

Programmi in Basic

Spett. Redazione,
sono un abbonato alla vostra rivista "Amiga Magazine". Questa è la seconda lettera che vi spedisco (la prima la spedii alla vecchia redazione) oltre, naturalmente, al coupon "Servizio lettori", che ritengo un mezzo utilissimo per lo scambio di informazioni tra il lettore e la rivista.
Ogni settimana mi reco in edicola per vedere le nuove riviste e fra tutti i "campioni" analizzati ritengo che una tra le migliori riviste per Amiga sia la vostra. Con la venuta della nuova redazione sono cambiate molte cose, la maggior parte delle quali in meglio.
La cosa che mi ha "deluso" è stata l'eliminazione degli articoli dedicati ai vari linguaggi per Amiga (AmigaBASIC, Assembly e C), anche se l'impostazione che la precedente rivista aveva dato a suddetti articoli era del tutto sballata, in quanto tali informazioni sono presenti su un sacco di manuali.
Per quanto riguarda l'Amiga BASIC sarebbe stato molto utile un articolo dettagliato su BOB o su come trattare le librerie (proprio su questi argomenti la documentazione è scarsa e spesso in inglese); ho apprezzato molto, per esempio, l'articolo sul comando PATTERN.
Un consiglio che vorrei darvi è, inoltre, quello di non soffermarvi molto sui giochi, bensì di analizzare anche programmi come compilatori, programmi DTV e DTP, ecc., anche se ho letto qualcosa come Sculpt Animate 4D sulla vostra rivista.
Veniamo ora ai pregi: innanzitutto la connessione a Compute! e a Transactor; la rivista ha assunto inoltre un aspetto molto più leggibile ed è diventata più ricca di informazioni e di notizie su prodotti, mostre, ecc.
Vi ammiro anche perché in Amiga Magazine c'è una completa simbiosi tra disco e supporto cartaceo e vi consiglio di insistere molto su questo punto e non separare la rivista dal disco.
Vorrei ora passare alle domande. Innanzitutto vorrei che pubblicaste un articolo sulla decodificazione del file .FD.
Inoltre, sarebbe molto utile pubblicare un inserto sull'uso delle librerie (come fatto su Transactor per quanto riguarda la graphics.library).
Sto inoltre studiando l'assembler 68000 e vorrei sapere se è utile per chi è ai primi passi l'acquisto del vostro libro "AMIGA Assembler" e del DEVPAC V. 2.14 (ho letto la recensione su Transactor).
Desidererei poi un programma (se possibile) scritto in Amiga BASIC o in Assembler che possa caricare file IFF HAM, IFF SMUS e IFF 8SVX (sarebbe bello avere come sfondi di giochi immagini HAM o suoni ed effetti sonori campionati).
Mi sembra di aver chiesto tutto ciò che desideravo sapere e, in attesa di vedere pubblicata sulla rivista la mia lettera (e magari le risposte alle mie domande), vi saluto cordialmente.

Giovanni Dicanio

Grazie prima di tutto per i complimenti e le critiche "costruttive" alla rivista che sono sempre gradite. Passiamo alle tue richieste: per

quanto riguarda i file .FD prossimamente troverai sulla rivista un programma in Basic per la loro decodificazione

Il libro "AMIGA Assembler" è stato studiato appositamente per coloro che vogliono imparare a programmare il 68000 e in modo particolare ad interagire con il sistema operativo di Amiga

Nel libro, e sul disco allegato, troverai moltissimi esempi di programmazione che sono stati sviluppati proprio per il DEVPAC della HiSoft

L'ultima tua richiesta è piuttosto utopistica

E', infatti, alquanto difficile scrivere un programma (specialmente in Basic) che faccia quello che hai richiesto.

Può comunque venirti in aiuto il programma SEUCK recensito alcuni mesi or sono su questa rivista.

Collaboratori ?

La redazione invita tutti i lettori ad inviarci i propri programmi e si riserva di pubblicare e retribuire quelli che ritiene i migliori. Sono ben accetti sia programmi di giochi, sia programmi di utility.

I programmi inviati non saranno, per ragioni redazionali, restituiti

Le opere dovranno essere inviate a

Gruppo Editoriale Jackson
Amiga Magazine
Via Pola, 9
20124 Milano

Dalla redazione...

Per ovvie ragioni di spazio non ci è possibile pubblicare tutte le richieste che giungono in redazione. Cercheremo di rispondere privatamente alle lettere non pubblicate.

a cura di S. Leemon, M. McLean
(© Compute! Publications, Inc.) e M. Anticoli

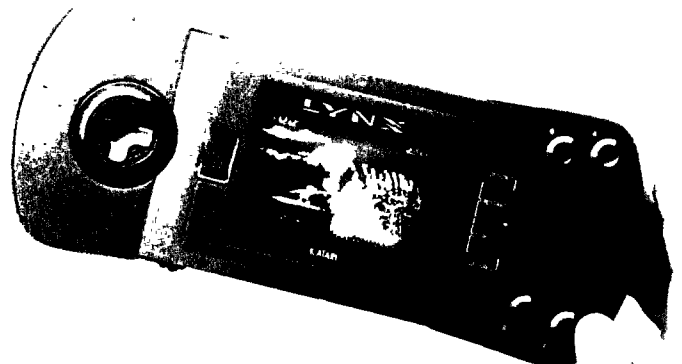
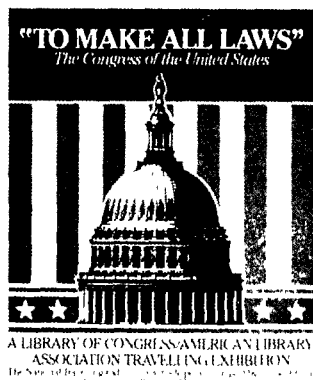
Il vincitore è...

Tetris della Spectrum Holobyte, che si è aggiudicato quattro dei premi SPA Excellence. Tetris, il primo gioco proveniente dall'Europa dell'Est, è stato giudicato Best Entertainment Program, Best Action/Strategy Program, Best Original Game e Best Consumer Program. Tra gli altri vincitori figurano: The Three Stooges della Cinemaware (Best Sound Achievement in a Non-Music Product), Battle Chess della Interplay (Best Graphics Achievement in a Non-Graphics Product) e Wayne Gretzky, della Bethesda Softworks (Best Sports Program).
Da parte nostra, attribuiamo alla SPA, il primo posto nella classifica "Il Premio dal Nome più Lungo"... ▲

Viaggio nel passato

Si discuteva tempo fa della possibilità di vedere un sistema Amiga di fascia bassa orientato ai videogiochi. Bene, sembra che i maghi della Commodore abbiano abbandonato il progetto, in favore di un nuovo computer a otto bit Non riuscendo ad abbassare radicalmente il prezzo di Amiga, stanno cercando di fare al Commodore 64 quello che Apple ha fatto al suo Apple IIe, trasformarlo in un "Commodore 64GS". La macchina avrebbe 128 Kbyte di RAM, espandibili a un Mbyte, un disk drive da 3.5" incorporato, e un chip grafico per il supporto di 256 colori contemporaneamente sullo schermo.
Naturalmente, sarà compatibile con il vecchio 64, per essere sicuri che nessuno scriva software per le nuove possibilità del computer
Mentre simpatizziamo con Commodore per il suo lodevole intento di rinnovare i suoi prodotti di fascia bassa, vogliamo sottolineare che qualcosa, in questa politica, va corretta. Non importano le sue specifiche, ma un computer a otto bit, oggi, sarà comunque obsoleto. Perfino Nintendo e Sega, ormai, realizzano console per videogiochi a 16 bit. E poi, chi sarà interessato a realizzare software per un altro computer di fascia bassa? Forse le software house della seconda ondata, quelle che non hanno ottenuto una licenza dalla Nintendo. Le case più celebri e consolidate, a nostro avviso, sarebbero molto meno interessate. ▲

Questo pazzo pazzo pazzo Tramiel

Abbiamo già parlato in passato di una macchina per videogiochi tascabile della Epyx, le cui caratteristiche venivano tenute segrete Bene, si tratta, come avevamo detto, di un apparecchio tascabile, dotato di uno schermo LCD da tre pollici da 160 x 100 pixel e di un processore 65C02 a 4 MHz capace non solo di supportare il movimento autonomo degli sprite, ma anche di ruotarli. A ciò si aggiunge una porta di I/O, per il collegamento di un massimo di otto unità per giochi multiplayer.
Per fornire alla macchina la necessaria forza di marketing, Epyx si è rivolta alla Atari di Jack Tramiel, che la produrrà e distribuirà. Questa notizia deve essere un po' amara per il presidente di Epyx, David Morse, e i suoi dipendenti provenienti dalla Amiga Quando Amiga Corporation si trovò in una situazione di carenza di liquidità, ottenne un prestito, di un milione di dollari, da Tramiel. Successivamente Tramiel acquistò la Atari, e Amiga Corporation entrò alla Commodore, appositamente per ripagare immediatamente Tramiel e così toglierli la possibilità di approfittare delle tecnologie di Amiga. Questa volta sembra che non sia andata altrettanto bene.
Il rapporto tra Epyx e Atari, comunque, non mancherà di presentare strani risvolti. Soprattutto per il fatto che la stazione di sviluppo per la nuova macchina è basata su Amiga 2000 Vi immaginate Jack Tramiel, proprietario della Atari, che compra Amiga per i suoi dipendenti? Un dirigente Epyx, interrogato sulla possibilità di utilizzare computer Atari al posto di Amiga, ha risposto "Certo. Se possono supportare grafica come questa, se hanno multitasking, se usano il Motorola 68020, se...". ▲

Amiga on the road

Commodore ha dato il suo supporto a una mostra itinerante organizzata dalla Library of Congress statunitense e dalla American Library Association, dedicata all'attività del Congresso degli USA dal 1789 al 1989 e destinata a toccare 30 città degli USA in un viaggio che durerà fino al Gennaio 1991 Amiga avrà il ruolo di un compagno interattivo per i visitatori, che potranno assistere a presentazioni audiovisuali della storia del Congresso in occasione del suo duecentesimo anniversario. ▲

Arriva l'aiuto per Deluxe Paint

RGB Computer & Video Creations presenta due nuovi programmi interattivi con funzioni di tutorial DeluxeHelp for DeluxePaint III include informazioni, consigli e demo delle opzioni di DeluxePaint III, il programma grafico della Electronic Arts. Il disco 1 tratta la grafica statica, mentre il disco 2 contiene dimostrazioni di tecniche di animazione utilizzate da animatori professionisti.
DeluxeHelp's Advanced Techniques - Textures & Backgrounds for DeluxePaint II, il primo di una serie denominata Advanced Techniques, è invece rivolto agli utenti medi o avanzati di DeluxePaint II. Il programma porta l'utente a comprendere i segreti della creazione di pattern, motivi grafici e sfondi usati in televisione, pubblicità e presentazioni. Tra gli effetti contenuti figurano legno, marmo, granito, oro e fumo. Ognuna delle 29 lezioni crea uno sfondo compiuto che può essere inserito in immagini fisse, animazioni, o applicato sulla superficie di oggetti tridimensionali. ▲

Battaglie tra Amigasauri

Potete tornare indietro nel tempo con il vostro Amiga, all'epoca in cui i dinosauri dominavano la Terra, con Dinowars della DigiTek Software. Questo gioco di strategia interattiva include sequenze complete di animazione e una minienciclopedia contenente una serie di nozioni sulle ere preistoriche Il gioco comprende otto varietà di dinosauri, divise in due fazioni che si affrontano per il controllo di un campo di battaglia preistorico. Come sempre, si gioca contro il computer o contro un avversario umano ▲

Che tempo farà

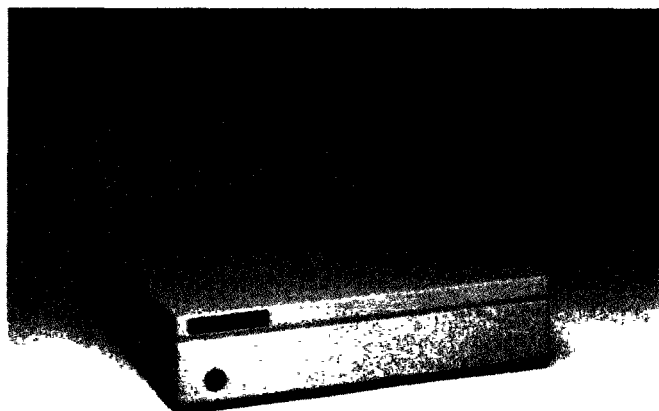
Sapevate che la più grande organizzazione privata di previsioni del tempo americana, la Accu-Weather dello State College della Pennsylvania, utilizza Amiga per creare carte del tempo e grafica per una parte dei suoi abbonati? La società dispone di un Amiga Weather Graphics System che permette alle televisioni locali di ricevere dati, previsioni e carte del tempo di alta qualità. Attualmente il sistema è usato da oltre 30 emittenti. Joel Myers, fondatore di Accu-Weather, dice che ha scelto Amiga perché economico e al tempo stesso congeniale per la produzione di grafica televisiva. Secondo Myers, il servizio di Accu-Weather è disponibile per le stazioni televisive a un terzo del costo abituale, grazie all'economicità di Amiga. ▲

Nuovi computer

La Nuova Voxson (nome già affermato in Italia e all'estero quale marchio di radio, autoradio, e televisori di alta tecnologia) è stata costituita nel 1988 con un Capitale Sociale di 8 miliardi ma diventò operativa solamente nei primi mesi del '90. Ha sede in un complesso di 12.000 metri quadrati, con linee di produzione interamente automizzate, in Via Flaminia Nuova nella zona Industriale di Roma Est. La Società ha iniziato l'attività nel campo dell'informatica con la produzione dimacchine di alta qualità e precisamente:

- una famiglia di terminali
- una famiglia di memorie di massa a disco magnetico, a nastro e a disco ottico.
- una serie di computer (a prezzi veramente esaltanti!) basati su microprocessori INTEL 286 e 386 con architettura XT/AT e MCA e sistemi operativi MS/DOS, OS/2 e UNIX.

A breve sarà aggiunta una famiglia di workstation, concepita per crescere dal semplice personal, via via, fino all'equivalente di un sistema MINI, in termini di potenza di calcolo e multiutenza opertiva.
E tutto ciò, secondo una strategia studiata per affrontare il mercato con un approccio nuovo e ben definito. La filosofia della Nuova Voxson non prevede più che sia l'utente ad adattare il proprio lavoro, la propria organizzazione al sistema informatico adottato; al contrario, hardware e software devono essere concepiti in modo da essere flessibili per qualsiasi funzione.
Occorrono per questo un complesso lavoro di progettazione e una tecnologia fortemente innovativa, ma, soprattutto, una mentalità nuova, sensibile a tutte le sollecitazioni che provengono dal mercato e capace di dare risposte ad ogni più piccola esigenza. ▲

Nuova Voxson
Via Flaminia Nuova, 513 - ROMA
Via Anco Marzio, 2 - MILANO
Tel.02/4696661-4695669 - Fax. 02/4696575

Nuova Sede

Computer Lab, famosissimo centro di assistenza autorizzato Commodore, ha cambiato sede. Il nuovo laboratorio, molto accogliente e spazioso, è sito in Via Cadore, 6 - Milano. ▲

Virus e cavalli di Troia

Il vostro Amiga ha dato ultimamente qualche segno di squilibrio?
Può darsi che sia stato contagiato da un virus. Sono ormai mesi che si parla di virus sulla stampa specializzata , e qualche notizia è perfino arrivata sui quotidiani.
A dispetto di tutti gli allarmi e di tutti i consigli, molti Amiga sono stati colpiti da virus. Attualmente le infezioni conosciute su Amiga sono più di 20, ognuna con un suo modo particolare di creare problemi. I due virus più noti sono: SCA (relativamente benigno) e Byte Bandit (distruttivo).
Nelle comunità di utenti Macintosh e IBM, i virus più diffusi sono i cosiddetti cavalli di Troia, che si nascondono all'interno di un file o di un programma. Gli utenti Amiga dovrebbero considerarsi fortunati, visto che a minacciare i loro computer c'è attualmente solo un cavallo di Troia, quello chiamato, impropriamente, IRQ Virus.
Ci sono molti modi per proteggersi da un'infezione, e almeno tre programmi in vendita a questo scopo. ▲

Amiga Virus Protection Toolbox
Abacus
Dept. L7
5370 52nd St. SE
Grand Rapids, MI 49512

Anti-Virus
DevWare
10474 Rancho Carmel Dr.
Rancho Bernardo, CA 92128

Virus Infection Protection
Discovery Software
163 Conduit St.
Annapolis, MD 21401

Novità

BIBLIOTECA DI BASE
SISTEMI E SERVIZI TELEMATICI
applicazioni WAN e LAN per l'azienda
Giuseppe Saccardi

L'evoluzione delle reti telematiche e la loro integrazione nell'ufficio

Giuseppe Saccardi
Si prende in esame l'evoluzione delle reti geografiche e delle reti locali mettendo in risalto la loro integrazione nella azienda
Cod.GT867 pp.240 L.32.000

Alfredo Sarich
Per tutti coloro che vogliono capire in concreto cos'è l'EDI, per essere pronti ad applicarlo al momento giusto e nel modo giusto.
Cod.GT838 pp.148 L.25.000

LA TELEMATICA NELL'UFFICIO
Riccardo Glücksmann
Tutte le applicazioni della telematica nell automazione dell ufficio

Riccardo Glucksmann
Gli aspetti essenziali della trasmissione dati, delle reti locali, dei centralini digitali e dei servizi telematici pubblici e privati
Cod.GT555 pp.256 L.35.000

U. Sgubbi, S. Farina, A. Gava
Per comprendere i principi di funzionamento dei dispositivi utilizzati nella trasmissione dati. Per studiare gli strumenti di misura telematici
Cod.GT814 pp.304 L.39.000

Riccardo Glucksmann
Un'analisi dei problemi teorico pratici di questa disciplina e i suoi sviluppi futuri in Italia e nel mondo
Cod.518D pp.286 L.28.000

SUL MEDESIMO ARGOMENTO

Giuseppe Saccardi
RETI LOCALI
Tipologie, caratteristiche e utilizzo
Cod.GTS478 pp.272 L.44.000

Giuseppe Saccardi
TRASMISSIONE DATI
Dispositivi standard e protocolli
Cod.GT783 pp. 304 L.45.000

Giuseppe Saccardi
RETI DATI
Caratteristiche, progetti e servizi telematici
Cod.617P pp.368 L.42.000

Mario Malcangi
IL MODEM
Teoria funzionamento e applicazioni
Cod.GTS479 pp.176 L.28.000

GRUPPO EDITORIALE JACKSON

I libri del Gruppo Editoriale Jackson sono in vendita presso le migliori librerie e computershop. Se ti è più comodo acquistarli per corrispondenza utilizza questo coupon

Da spedire in busta chiusa a **GRUPPO EDITORIALE JACKSON, Via Rosellini 12 - 20124 Milano**
Si, inviatemi i volumi sottoelencati

INDICARE CHIARAMENTE CODICI E QUANTITA' DEI VOLUMI RICHIESTI

Codice	Q.ta	Codice	Q.ta	Codice	Q.ta	Codice	Q.ta	Codice	Q.ta

Ordine minimo L. 60.000 + L. 4.500 per contributo fisso spese di spedizione

❑ Sono titolare della Jackson Card '90 n. ______ e ho diritto allo sconto del 10% (fino al 31/12/90)
❑ Non sono titolare

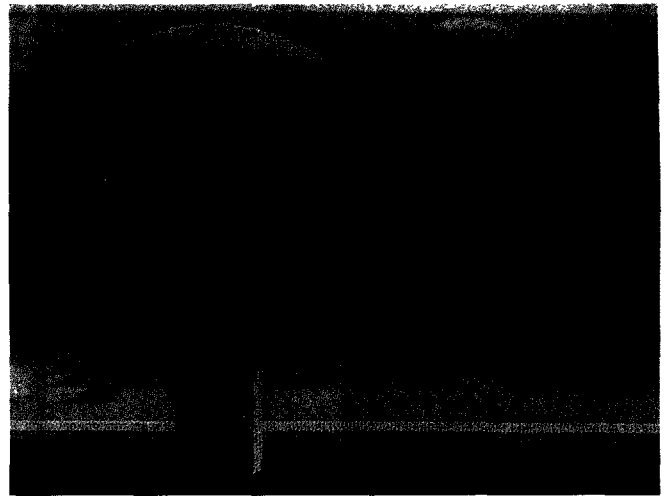
MODALITÀ DI PAGAMENTO. ❑ Contro Assegno postale al ricevimento dei volumi
❑ Assegno allegato n. ______ Banca ______
❑ Ho effettuato il pagamento a mezzo ❑ Versamento sul c/c post. n° 11666203 a Voi intestato e allego fotocopia della ricevuta
❑ Addebitatemi l'importo di L. ______ sulla carta di credito ❑ Visa ❑ American Express ❑ Diners Club ❑ Carta Si
Conto n. ______ data di scadenza ______
❑ Richiedo fattura (Partita IVA n. ______)

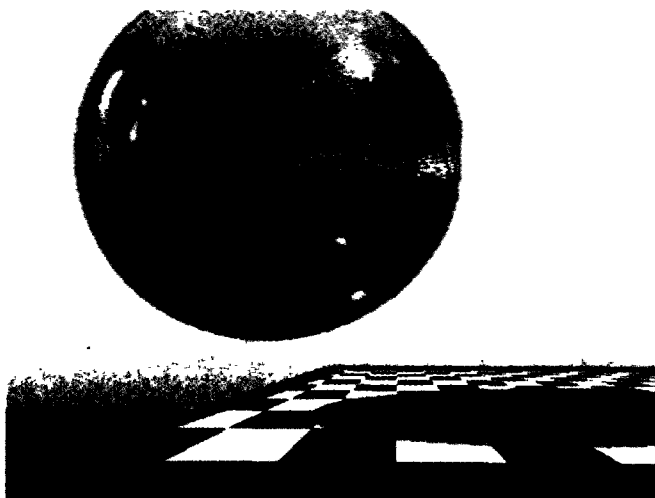
Cognome e Nome ______
Via ______ n° ______
Cap ______ Città ______ Prov ______
Tel ______ Data ______ Firma ______

TM 1/11

Art Gallery

Amiga Art Gallery è una rubrica mensile aperta a tutti gli Amigartisti.
Per partecipare è sufficiente inviare una fotografia, diapositiva o dischetto, con l'immagine salvata in formato IFF, della propria opera con qualche riga di descrizione come: nome dell'autore, programma con cui avete creato l'opera, oppure il metodo utilizzato ecc.
Ricordiamo però che Art Gallery non è né una gara né una competizione !!
Le opere pubblicate vengono scelte a giudizio insindacabile della redazione, le opere pervenute non vengono restituite.
Se veramente vi ritenete degli artisti con Amiga, cosa aspettate a inviarci le vostre opere, magari scopriamo un nuovo Picasso !!

Le opere vanno inviate a:
Gruppo Editoriale Jackson
Amiga Magazine
Area Consumer
Rubrica "Art Gallery"
Via Pola, 9
20124 Milano

Alba
Roberta Radaelli
Programma: Deluxe Paint III

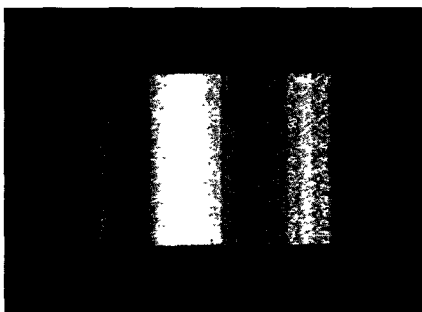
Mosaico
Roberto Gazzola
Programma: Deluxe Paint II

Il furturo degli scacchi
Giuseppe Tiozzo
Programma: Deluxe Paint III

Arti vive

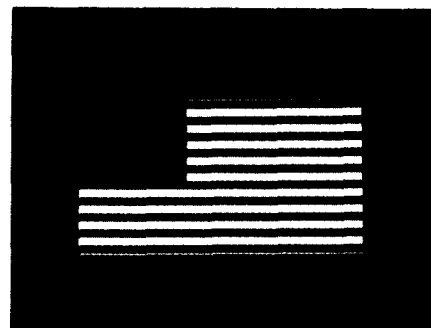
La bandiera dell'arte Amiga vola sempre più alta. Questa prova pratica di programmi per disegnare e di strumenti grafici può aiutarvi ad aggiungere il vostro vessillo alla parata degli artisti di Amiga.

Lee Noel, Jr.

Per disegnare la bandiera siamo partiti da questa immagine di base

poi l'abbiamo avvolta su questo rettangolo per aggiungerle le ombre necessarie

Una rassegna completa sull'attuale "stato dell'arte" dell'arte Amiga potrebbe facilmente riempire un libro. Ma bisogna cominciare a fissare un limite da qualche parte. Così, in questo articolo, mi occuperò solamente degli equivalenti elettronici delle arti visive bidimensionali. Ciò significa che tratteremo di materiali che potrebbero normalmente essere rappresentati su tela, carta o un'altra superficie piatta (come è tipico di pitture a olio e acquerelli, acquaforti, disegni, fotomontaggi, collage e così via), ma che preferiamo far vivere sul monitor di Amiga.
Concentreremo la nostra attenzione anche su argomenti e concetti di carattere generale, che aiuteranno a fornire all'artista principiante una visione generale di questo mezzo emozionante. Per fornire dettagli e conoscenze su tecniche e strumenti specifici, tre noti artisti hanno dato il loro contributo negli articoli di contorno: in essi appaiono al lavoro su diversi tipi di effettivi progetti artistici realizzati con Amiga.

Pitture rupestri

Una volta le cose erano chiare e prive di confusione nel mondo della grafica per Amiga. Quando apparve la versione originale del computer (ora chiamata 1000, ma allora conosciuta semplicemente come Amiga), gli artisti avevano accesso a una palette massima di 32 colori, posti simultaneamente sullo schermo, e a un certo numero di risoluzioni, che andavano da quella bassa (320 x 200 pixel) [256 PAL, N.d.T.] a quella alta (640 x 400 pixel) [512 PAL, N.d T.].

Le risoluzioni più alte erano molto più elevate di quelle precedentemente offerte da computer a prezzi popolari, e la gamma totale dei 4096 colori, presente su Amiga, sembrava un arcobaleno senza limiti. La palette di Amiga è il risultato del metodo RGB (Red, Green e Blue - rosso, verde e blu) che il computer usa per creare un dato colore dello schermo e che consiste nel mescolare differenti intensità dei tre colori fondamentali della luce. Ciascun colore può essere regolato su un campo di 16 valori, ottenendo una gamma totale di 16 x 16 x 16 (cioè 4096) colori. Siccome ogni singolo pixel in uno schermo di Amiga può essere colorato indipendentemente dai suoi vicini, e siccome la palette di un'immagine potrebbe essere costruita con colori presi in qualsiasi punto della scatola dei colori del computer, non ci volle molto prima che un torrente di lavori grafici di grande effetto cominciasse a fluire dalla macchina.
Almeno, tutti gli altri furono impressionati. A me le immagini parvero vistose e piene di grumi. Ancor peggio, l'ampia maggioranza di questi primi lavori era in 32 colori e in bassa risoluzione. Francamente, questo mi lasciò deluso. Non solo Amiga

era relativamente caro, ma veniva usato principalmente per produrre disegni che il Commodore 64 poteva eguagliare. Riconosco uno schermo 320 x 200, quando lo vedo, e mi aspettavo qualcosa di meglio da questo computer potente, nuovo e orientato alla grafica.

Risoluzioni oscillanti

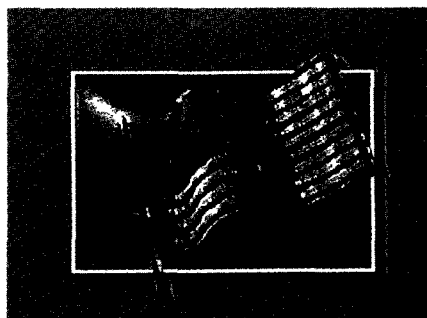
Mi domandavo perché non si usavano le risoluzioni più alte. Una piccola, molto piccola, sperimentazione rapidamente rivelò il perché. La risoluzione immediatamente più alta, 640 x 200 [256 PAL, N.d.T.], produceva orribili pixel, allungati verso l'alto; buoni forse per creare grafica alla El Greco, ma fonte di notevoli limitazioni, in quanto una linea orizzontale era costretta dalle regole della geometria ad avere uno spessore tre volte più grande di una verticale. Essenzialmente, questo significava dimenticare quasi, in questo modo grafico, il disegno a tratto più fine.
Pixel più gradevoli, più vicini alla classica proporzione aurea dei rettangoli, venivano prodotti quando ci si muoveva in su di un gradino nella scala delle risoluzioni di Amiga, ma il prezzo da pagare per vederli era troppo assurdo per me. Era il modo interlacciato, la risoluzione 320 x 400 [512 PAL, N.d.T.], ed era chiaramente stato progettato con un occhio alla novella Steppenwolf di Hermann Hesse. In quel libro c'è uno spettacolo piuttosto insolito, con un biglietto d'ingresso piuttosto inusuale. Al di fuori l'insegna recita "Prezzo d'ingresso, la tua mente". Questo, letteralmente, era il prezzo che dovevo pagare per l'uso del modo interlacciato. Lavorando con il furibondo tremolio di questo modo grafico, è possibile sviluppare un mal di testa del grado 9 della scala Richter, senza l'aiuto di traumi cranici o malattie. A differenza dell'uomo-bestia Steppenwolf, questo era un prezzo che io non ero preparato a pagare.
La risoluzione piu alta di Amiga, 640 x 400 [512 PAL, N.d.T.], risultava essere un altro modo interlacciato ed era, per giunta, anche avido di memoria Offrendo ridotte possibilità di colori e affaticando abbastanza la vista da riuscire a distaccare anche la retina più saldamente ancorata, questo modo grafico trasformava lo schermo di Amiga in un pannello vibrante sul quale i prodotti grafici assomigliavano generalmente a tele scomposte costruite da ragni molto agitati.
Completata la mia deludente ispezione dei modi grafici di Amiga, lo classificai (nella mia mente, almeno) come una macchina buona per i giochi, ma assolutamente inadatta ad imprese grafiche serie.

L'immagine finale, con lo sfondo, è stata prodotta con Photon Paint 2 0

L'HAM introduce la pittura

Naturalmente, solo poche settimane dopo (attualmente non ho un chiaro ricordo del momento. Sinceramente Vostro, R. Reagan) stavo allegramente usando Amiga per produrre grafica di tutti i tipi. La mia inimmaginabile conversione era stata accompagnata dall'apparizione di programmi che facevano uso del misterioso modo grafico di Amiga Hold-And-Modify (HAM). L'HAM consentì ad Amiga di giocare un ruolo più sostanzioso nelle arti grafiche e servì a riconfermare la mia fede in un lato ironico di tutta l'esistenza.
L'HAM infrange la barriera dei 32 colori, creando uno schermo in cui tutti i 4096 colori del computer possono apparire contemporaneamente, fatta salva qualche limitazione E' un modo grafico difficile da programmare e difficile da capire e, secondo una di quelle storie apocrife sullo sviluppo di Amiga, ci fu un tempo in cui i progettisti intendevano eliminarlo del tutto dalla macchina. Fortunatamente non lo fecero, ma sembra che ci sia voluto del tempo in più perché gli sviluppatori del software arrivassero a dominare e usare questa complessa configurazione del display
In parole povere, ecco come lavora l'HAM. Una palette di 16 colori di base viene fissata per un certo schermo I componenti RGB per questi colori costituiscono le fondazioni su cui l'HAM costruirà il resto dei colori da includere nell'immagine. A seconda della richiesta del programma di disegno o di digitalizzazione in uso, un singolo pixel può essere colorato direttamente con uno dei 16 colori della palette o, molto più probabilmente, il pixel verrà colorato mediante calcoli effettuati nel rispetto delle regole del modo HAM. Nell'ultimo caso, viene scelto un colore finale per il pixel e il programma tenta di soddisfare quel valore. Ciò viene fatto esaminando il pixel posto alla sinistra di quello da colorare In modo HAM, il software ha la possibilità di conservare (hold) due dei tre valori RGB che determinano il colore del pixel di sinistra e simultaneamente di modificare (modify) uno di loro. Questo insieme modificato di valori RGB viene poi usato per colorare il pixel di destra.

Un effetto calcolato

Sotto un certo aspetto, è come se l'HAM offrisse un insieme completo di cursori RGB per ciascun pixel sullo schermo. Due dei tre cursori devono essere uguali a quelli del pixel di sinistra, ma uno è completamente indipendente. In questo modo, in un esempio spesso citato, ci vogliono quattro pixel HAM per andare dal nero al bianco.
In pratica, bianco e nero sono spesso inclusi nei 16 colori della palette

Profilo di un artista RICHARD NICHOL

L'immaginazione e Amiga sono sempre andati mano nella mano e i programmi di ray tracing forniscono gli strumenti per proiettare plasticamente la vostra immaginazione sullo schermo.
"Art Museum" (museo d'arte) e "Grecian Landscape" (paesaggio greco) sono stati generati con Turbo Silver 3.0 e ritoccati con Deluxe PhotoLab. Ho creato gli oggetti con Modeler 3D, li ho convertiti nel formato corretto con Interchange e importati in Turbo Silver per l'elaborazione. In "Art Museum" le cornici delle pitture e i muri di marmo sono stati digitalizzati usando Framegrabber e poi usati come pennelli assieme alle pitture nella scena. I riflessi del pavimento apportano un tocco di realismo all'immagine.
In "Grecian Landscape" la statua greca è stata digitalizzata e tracciata sopra una maschera con la stessa forma. Ho creato le montagne in distanza con Terrain Generator della Impulse e le ho poi dipinte con del marmo digitalizzato. Turbo Silver fornisce una flessibilità quasi illimitata nel definire le caratteristiche delle superfici e, nel caso dell'acqua, ho impiegato circa 30 elaborazioni ad ottenere la combinazione di trasparenza, rifrazione e riflessione che volevo. Le idee per i miei lavori provengono da riviste e libri storici. Inizio con un tema, per dire l'arte giapponese o l'architettura romana, e poi scelgo il tipo di oggetti che potrebbero essere creati per rappresentare quel tema. Inoltre, cerco immagini che posso digitalizzare per aggiungere il tipo di realismo che l'elaborazione 3-D, da sola, non potrebbe fornire.
Usare Amiga come strumento artistico mi consente di muovere oggetti in uno spazio tridimensionale e di provare diversi angoli di visuale, profondità di campo e illuminazione al fine di ottenere una composizione bilanciata dell'immagine. Questo è un enorme beneficio se confrontato con i mezzi tradizionali disponibili all'artista.
Un altro elemento chiave nella produzione di immagini ray tracing visivamente affascinanti è un uso generoso delle superfici riflettenti o trasparenti. E qui che i calcoli matematici del ray tracing trasformano la tua immagine in qualcosa di magico e aggiungono il realismo fotografico che può essere così spettacolare. Alla fine, le vostre lunghe ore di lavoro e quelle di Amiga vi saranno ripagate sotto forma di oggetti artistici sorprendenti.

Art Museum

Grecian Landscape

di base per le immagini HAM. La palette è immune dalle regole HAM, così bianco e nero si trovano spesso adiacenti in immagini HAM.
Chiaramente, l'effetto di blocco dei cursori significa un mucchio di lavoro per il programma che opera in questo modo. Gli schermi HAM sono prodotti come il risultato di una serie ricorrente di rapidi calcoli. Ogni volta che un colore è aggiunto o rimosso, i colori finali in un'area particolare cambiano, e il software HAM deve ricalcolare i componenti RGB che non solo tentino di soddisfare i nuovi valori finali, ma che lascino sufficientemente stabili anche i dettagli già esistenti sullo schermo. Sebbene tecnicamente l'HAM sia una specie di palude, può produrre immagini che sono molto piacevoli, persino per gli occhi stremati di questo osservatore (un tipico dettaglio bizzarro: i 16 colori della palette HAM di base potrebbero essere il bianco, il nero e i 14 possibili toni di grigio del computer; questa palette potrebbe essere usata per produrre immagini piene di colori, in cui non compaia nessuno dei colori di base). Un'immagine HAM nella risoluzione 320 x 200 [256 PAL, N.d.T.] ha l'aspetto di un'immagine di qualità televisiva solo leggermente più sgranata. Le immagini HAM alla risoluzione di 320 x 400 [512 PAL, N.d.T.] sono realmente molto vicine allo standard TV, sebbene questa sia una delle temute risoluzioni interlacciate.
Perché questi riferimenti alla televisione? La risposta è semplice: questo mezzo di comunicazione onnipresente ha messo milioni di schermi a raggi catodici in milioni di case, da decenni. Quale miglior modo per valutare le immagini degli schermi dei computer, che misurarle con

La cassetta degli atrezzi

Programmi per disegnare

- Aegis Images
- DeluxePaint III
- Deluxe PhotoLab
- Digi-Paint
- Photon Paint 2.0

Programmi per l'elaborazione delle immagini

- Butcher
- PIXmate

Tutti i programmi per disegnare selezionati forniscono un ragionevole livello di prestazioni e contengono gli strumenti ormai standard per disegnare linee rette o curve, oltre a rettangoli, ellissi e poligoni di vario tipo, sia pieni che vuoti. Per il lavoro sui dettagli, comprendono tutti un raffinato sistema per ingrandire, di molti ordini di grandezza, una piccola area. Inoltre, ognuno di questi pacchetti fornisce un'ampia gamma di pennelli, di funzioni per il riempimento delle aree e per la sfumatura dei colori, assieme ad altri effetti pittorici

questo parametro così familiare? I correnti standard televisivi non sono ad alta risoluzione, ma a causa della possibilità di riprodurre virtualmente ogni colore in ogni pixel, essi possono offrire spesso immagini visivamente accettabili quasi di ogni più piccolo particolare. Allo stesso modo, i colori addizionali dell'HAM consentono ad Amiga di trascendere la sua risoluzione intrinseca, approssimando i dettagli attraverso la manipolazione dei colori. Come la TV, Amiga riesce a visualizzare in HAM immagini quasi fotografiche. Una delle prime principali apparizioni dell'HAM fu nel classico digitalizzatore video Digi-View della NewTek, che traduceva (e continua a tradurre) qualsiasi cosa fosse inquadrata da una telecamera in schermi Amiga molto belli e simili a fotografie.

Nel raggiungere una qualità quasi televisiva, il modo HAM espande la gamma della grafica di Amiga in un regno più maturo, dove gli equivalenti elettronici degli effetti tradizionali, come acquerello e aerografo, sono finalmente più di semplici vuoti termini usati nella documentazione dei programmi. E, dal momento che l'HAM è ancora un fenomeno relativamente inesplorato, è probabilmente una cosa certa che sarà fatto sempre di più per migliorare la qualità del giovane modo grafico.

Né in cielo né in terra

Una breve menzione dovrebbe essere fatta del rimanente modo grafico disponibile, l'Extra Half-Brite. Con esso il computer può visualizzare simultaneamente sullo schermo fino a 64 colori. Sfortunatamente, 32 di questi colori sono solo versioni a intensità dimezzata dei colori presenti nei 32 registri regolari dei colori. Se cambiate il colore in uno dei registri principali, cambia anche l'associato colore half-brite. Questo modo è considerato come un modo eccellente per produrre ombre. Per me, è un'insoddisfacente approssimazione dell'HAM. Inoltre non è consentito dall'hardware su alcuni dei più vecchi Amiga 1000, pertanto è un modo che dovrebbe essere avvicinato con cautela.

Quo vadis?

Il grafico che inizia con Amiga si trova di fronte alcuni problemi: con quale risoluzione lavorare? Quanti colori usare? E' veramente così brutto il modo interlacciato? E' veramente così grande l'HAM? Ancora una volta, è la versatilità di Amiga che causa queste domande, ma per un artista queste domande rappresentano un delizioso e fluido mezzo espressivo. Tuttavia, il modo con il quale un artista risponde ad esse tende a stabilire delle scelte precise. Naturalmente, su Amiga, queste risposte sono implicite nella scelta del software grafico operata dall'artista. Se volete lavorare in HAM, ma avete un programma a 32 colori, siete evidentemente proprio sfortunati. Ovviamente, il software è la chiave dell'arte su Amiga.

Dal momento che scegliere il programma giusto è oltremodo cruciale nell'arena delle arti visive, ecco una rassegna di alcuni prodotti ben affermati nel campo. Ho chiaramente in testa la lettera di Wilson Lee nel primo numero di Amiga Resource. A proposito dei programmi per disegnare, egli chiedeva: "Potreste aiutarmi a scegliere un programma per disegnare con Amiga che sia completo, non destinato solamente agli utenti professionali e che abbia un prezzo ragionevole?" Spero di poter essere d'aiuto.

I pacchetti che abbiamo selezionato consistono in cinque programmi per disegnare e due pacchetti per l'elaborazione delle immagini. Quest'ultima coppia di programmi non solo aiuta l'artista a migliorare e a modificare il proprio lavoro in modi diversi, quasi magici, essa fornisce anche una strada attraverso la quale modi e risoluzioni possono viaggiare, quando vengono trasformati da una configurazione in un'altra.

Aegis Images

Images è all'incirca il modo più economico per entrare nel mondo delle arti grafiche di Amiga, avendo ancora a disposizione del software potente.

Prima di prendere in considerazione l'acquisto di questo programma, tuttavia, ci sono tre fattori da considerare. Primo, Images offre un accesso a solo due tipi di risoluzione, 320 x 200 e 640 x 200, dunque non permette nessun modo interlacciato. Neanche il disegno in HAM o in Extra Half-Brite è incluso. Infine, l'interfaccia utente di Images e alcuni dei concetti che ha fatto propri sono decisamente differenti da quelli che si possono trovare nella corrente principale delle applicazioni Amiga (a seconda della sensibilità personale, questa può anche non essere una cattiva cosa).

Oltre a possedere una generosa selezione dei tipici strumenti da di-

segno e da pittura, Images eccelle nella creazione di vari tipi di figure geometriche. Attraverso il menu Shape, il programma offre sottomenu specializzati di opzioni che possono essere usate dal grafico per stabilire diverse categorie di intersezioni e condivisione dei contorni per qualcuna o per tutte le figure di un certo tipo.

Un altro aspetto attraente di questo programma è il menu Fast. Questo piccolo e mobile menu permette un facile accesso all'intera palette e alle ultime quattro maschere usate per il riempimento delle aree. Espandendo il menu Fast, si può, con estrema facilità, selezionare Undo, Options, uno degli ultimi tre pennelli usati o uno degli ultimi tre strumenti usati. Il menu Fast offre comodità senza disordine ed è un beneficio reale per coloro che preferiscono vedere la maggior parte dello schermo su cui stanno lavorando.

C'è anche un editor per le maschere di riempimento, facile da usare. Anche se, oggi come oggi, questi editor per le maschere sono spesso

Profilo di un artista BILL MELENDEZ

Disegnare delle immagini su Amiga è un'esperienza analoga alla esecuzione di acquerelli o anche di pitture a olio. Ogni metodologia richiede che scelga una palette prima di iniziare a dipingere. I colori da usare sono determinati fino a un certo punto dal soggetto che ho deciso di ritrarre. Per rendere le cose più semplici, uso palette preesistenti che si accordano con lo stato d'animo che voglio esprimere.

Così facendo, elimino del tempo inutile o il fatto di avere i colori nella posizione sbagliata della palette. Scelgo il soggetto, come nel caso della ragazza peruviana, a seconda di quanto l'immagine voglia essere provocante e a seconda del fascino che il soggetto esercita su di me. Immagino anche come apparirebbe un soggetto, quando venisse posto in un'immagine. Pianificare la composizione, i colori e la quantità di dettagli desiderati mi aiuta ad evitare progetti che vadano oltre le possibilità di Amiga.

"Perù Ad" (pubblicità del Perù) venne fuori dai miei file di immagini, come una serie di immagini per rivista. La piccola ragazza mi colpì come un soggetto ideale. La posa, l'aspetto e l'innocenza sembrarono ideali per il video. Di tutte le immagini, faccio per prima cosa i disegni delle linee e poi li combino tra loro come in un fotomontaggio. Tutto ciò viene realizzato tracciando i disegni sul computer con la tavoletta grafica Easyl e Deluxe Paint II. Ciascun disegno viene spostato come un pennello e poi riversato sul fotomontaggio.

Completato il disegno, comincia il processo di aggiunta dei colori. Nel modo ad alta risoluzione di DeluxePaint (640 x 400), la palette è limitata a 16 colori. Con l'immaginazione e i disegni di tratteggio incrociato, posso incrementare il numero di colori disponibili sullo schermo. Sia "Perù Ad" che "Fish" (pesce) usano una tecnica di disegno a scacchiera per incrementare il numero di colori percepibili. Questo metodo è simile a quello usato dalle riviste per la stampa delle immagini a colori.

Il metodo che uso per aggiungere colore a un disegno è un approccio color-by-numbers.

Perù Ad

Fish

Usando lo strumento per il riempimento di DeluxePaint, riempio il disegno del colore principale e delle ombre. Tra il colore principale e le ombre inserisco i disegni a scacchiera. Il pennello permette al colore principale di apparire a pixel alternati. Normalmente il pennello a scacchiera usa un colore più chiaro di quello su cui viene inserito. Uso lo strumento rubber-band per espandere ulteriormente l'intrusione di ombre più chiare in quelle più scure dell'immagine.

Per ottenere colori addizionali dispongo delle linee diagonali dello spessore di un pixel, lungo i contorni di un'area di colore principale. Alterno i colori, mentre inserisco ogni linea trattata con il rubber-band al suo posto. Il procedimento è lento, ma i risultati sono interessanti. Entrambe le immagini furono realizzate come parte di un demo per aiutare i rivenditori locali a vendere Amiga. Il demo contiene grafica in alta risoluzione aggiuntiva che è stata combinata assieme al resto al fine di formare uno slide show con Lights!Camera! Action !

considerati fuori moda, penso che siano utilissimi per lavori di decorazione con bordi o piccoli elementi ripetuti.
Il metodo del programma per il "taglia", "copia" e "incolla" si fonda sull'insolito dispositivo Frame.
Ogni segmento dello schermo catturato dai bordi del Frame può anche essere usato come pennello, ridimensionato trascinando il margine del Frame o ruotato con grande velocità.
Sebbene Images contenga molte funzioni per ombreggiare e sfumare i colori, il suo forte sta probabilmente nei lavori con molti spigoli e con motivi ripetuti. E' un pacchetto divertente con cui imparare e con cui lavorare.
La sua peculiare interfaccia utente non vi preparerà per i formati più standardizzati che si trovano nella maggioranza degli altri programmi per disegnare, ma penso che l'interfaccia Images possa attualmente risultare molto intuitiva e adatta alla grafica per molte persone

Digi-Paint

Digi-Paint fu il primo programma per disegnare in HAM, ed è il mio programma preferito.
Spesso considerato come un editor grafico di base, complementare a Digi-View e da questo scorporato, il programma originale ha una semplicità così lineare che essa tende a offuscare la sua notevole potenza
Io considero Digi-Paint un programma per disegnare quasi ideale.
Oltre agli equivalenti HAM dei soliti strumenti e pennelli, Digi-Paint fornisce anche funzioni per il riempimento e per la sfumatura che sono le migliori esistenti.
Camuffato come semplice pulsante Repeat, il programma fornisce all'artista la possibilità di eseguire l'Undo di un'azione e poi di ripeterla esattamente, dopo aver modificato opzioni e parametri.
Ciò può non sembrare granché, ma in connessione con i precisi controlli permessi dal software, consente operazioni dalla natura più sottile.
Questo è il programma da usare per generare equivalenti degli acquerelli più leggeri o dei lavori ad aerografo più delicati.
Con esso, potete riprodurre effetti che vanno dalla luce del sole attraverso un vetro colorato, all'indaco perfetto di un cielo notturno visto molto tempo fa, ma mai dimenticato.
Come osservazione meno nauseante, potrei affermare che ritengo questo il programma più adatto alla manipolazione di immagini HAM digitalizzate.
Altri, senza dubbio, non saranno d'accordo, ma il modo ingradimento di Digi-Paint offre il più sicuro controllo manuale che io abbia mai trovato, per l'editing di immagini HAM pixel per pixel.
Una nuova potente versione del programma, Digi-Paint 3, sarà probabilmente disponibile nel momento in cui leggerete questo articolo. Sembra che il programma includerà alcune nuove caratteristiche, compresi menu e controlli grafici molto migliorati [esso ha introdotto anche un nuovo rivoluzionario modo grafico detto "Dynamic HiRes" che permette di usare l'HAM in alta risoluzione e in overscan (fino a 780 x 480 pixel in NTSC), N.d.T.].
Anche senza vederlo, Digi-Paint 3 è il programma HAM che comprerei (vedere Amiga Magazine 11 per una recensione completa).
Ogni grafico ha una sensibilità individuale. E' importante tentare di trovare il software che sposa la vostra, così continuate a leggere!

Deluxe PhotoLab

Deluxe PhotoLab consiste di tre elementi. Paint, Colors e Posters. In questo contesto, noi ci occuperemo solo del primo, ma gli altri due sono utili complementi, in quanto permettono di manipolare, elaborare e poi stampare su macchine diverse i prodotti di Paint.
Questo è un pacchetto importante, e non sono convinto che abbia ottenuto il riconoscimento che meritava
Paint di PhotoLab è il solo software che permette al grafico di avere accesso a tutti i modi grafici di Amiga da un solo programma. Non solo tutti i modi sono accessibili, ma potete anche lavorare su di essi contemporaneamente. Ciascuna risoluzione o modo appare nella propria finestra e potete trasferire elementi avanti e indietro tra le finestre con facilità
Paint di PhotoLab è un programma grafico dotato di tutte le funzioni, indipendentemente dal modo grafico con cui lavora. I suoi aspetti migliori stanno nei molti modi di pittura e ombreggiatura, che comprendono operazioni logiche (AND, OR e così via, che possono essere usate per miscelare e modificare elementi degli schermi in molti modi utili e creativi), che sono particolarmente utili in quanto questo programma include la propria versione della funzione Repeat di Digi-Paint.
Una caratteristica unica è la capacità di creare, testare e conservare molti colori aggiuntivi in un "Paint set" nella finestra della palette. Questi colori addizionali sono indipendenti da quelli che vengono controllati dalla palette dello schermo. Così, possono essere provati senza temere di disturbare il lavoro già fatto I colori del Paint set, ritenuti adeguati, possono essere applicati direttamente alle immagini HAM; i prodotti non HAM sono colorati con i colori della palette normale che si avvicina di più alla selezione del Paint set.
Per coloro che vogliano accedere a tutti i modi grafici di Amiga, ma non desiderino imparare ad usare molti

Creata con Paint di Deluxe PhotoLab in modo HAM (4096 colori), 320 x 200

programmi, questa è l'unica scelta possibile.
Deluxe·PhotoLab contiene così tante cose che il suo prezzo è attualmente un affare (per inciso, le caratteristiche e la documentazione di Paint sono analoghe a quelle del suo compagno di scuderia della Electronic Arts: DeluxePaint III; ciò può essere utile agli artisti che vogliano usare entrambi i programmi).

Deluxe Paint III

Rhett Anderson ha recensito questa nuova versione del vecchio e fidato programma, nel numero 9 del 1990 di Amiga Magazine Egli afferma: "E' eccezionale, forse il migliore programma esistente per microcomputer".
Io, il solito incontentabile, vorrei dissentire. Non ho apprezzato nessuna versione di questo programma e non riesco a trovare un motivo per fare eccezione per l'ultima. Fatemi spiegare, ma ricordate che le mie opinioni sono determinate da gusti personali in fatto di software e prodotti artistici. Quasi tutti, nell'universo, concordano con le iperboli di Anderson.
Non solo DeluxePaint III non lavora in HAM, ma la sua interfaccia utente è proprio orrenda. Alcune delle icone per la selezione degli strumenti offrono indizi visivi molto poveri sugli strumenti che rappresentano. L'icona del Brush Selector è la peggiore del mucchio, ma si riferisce a una delle funzioni più importanti del programma. E persino Anderson ammette che DPaint III non è molto adatto all'uso su una macchina con 512K.
Per essere onesti, il programma è carico di ogni immaginabile strumento e funzione. Gonfio potrebbe essere una parola più adatta di carico; questo programma è veramente affollato.
Come conseguenza, esso presenta la curva di apprendimento più ripida di ogni altro pacchetto qui esaminato. Finestre, gadget e requester giacciono a molti strati di profondità e molti di essi usano un simbolismo e un gergo imperscrutabili. Ma una volta che l'artista cominci a penetrare profondamente nella sua struttura, i frutti compenseranno ampiamente gli sforzi.
Dal momento che DeluxePaint III contiene ogni strumento per disegnare, colorare, ombreggiare o mascherare conosciuto, ha poco senso l'elencarli uno per uno.
La maggior parte di queste funzioni sono più o meno identiche a quelle presenti in DeluxePaint II, ad ogni modo. L'impulso principale che ha mosso il creatore di DPaint III, Dan Silva, è stato il desiderio di aggiungere l'animazione al pacchetto. Come al solito, vi è riuscito in maniera straordinaria.
Ci sono tre modi differenti per animare le creazioni grafica. Le animazioni possono essere salvate e riprodotte, naturalmente. C'è persino un'applicazione autonoma che potete mettere nel vostro disco di animazioni in modo che chiunque abbia un Amiga possa vedere le vostre creazioni cominciare a vivere e danzare attraverso i pixel.
Se vi piace la grafica Amiga non HAM, volete il più completo programma in circolazione e vi piacerebbe porre le mani in un sistema di animazione ben implementato (non HAM, non dimenticatelo), DeluxePaint III ha tutto quello che vi serve.

Creati con Photon Paint 2.0 in modo HAM (4096 colori), 320 x 200, questi oggetti d'arte non sono altro che un singolo pennello ricavato da un occhio, rozzamente disegnato, che è stato poi avvolto su alcune sagome predefinite Lo sfondo è stato ottenuto con il riempimento ad un solo colore e le sue sfumature sono state realizzate con i puri controlli sull'ombreggiatura forniti dal programma

Creati con Photon Paint 2.0 in modo HAM (4096 colori), 320 x 400, queste picchi frastagliati sono stati fatti avvolgendo un piatto pennello giallo sul rettangolo con vari toni di grigio che si mostra al di sotto delle montagne Le differenze di intensità nell'area grigia diventano differenze apparenti di altitudine nel pennello che viene avvolto su di essa L'illuminazione è stata predisposta per cadere sulle cime provenendo da destra

Photon Paint 2.0

Questo è un altro grande programma HAM e una versione piuttosto recente del Photon Paint originale. In qualche modo, questo programma offre lo stesso livello di complessità di DeluxePaint III, ma la sua interfaccia è molto più semplice da capire. Da questo punto di vista, devo aggiungere un forte voto di approvazione per la meravigliosa documentazione dell'ex-alunna di Don Bluth, Heidi Turnipseed. Scrive sempre per l'artista, non per il maniaco del computer. Le sue osservazioni sono sufficientemente esaurienti da illuminare anche il principiante più nervoso; i suoi tutorial sono sviluppati con cura e offrono una chiara comprensione della metodologia di lavoro di un artista.
Ancora una volta, questo è un software con troppe caratteristiche per poterle elencare. Per darvi un assaggio della profondità di Photon Paint 2.0, descriverò uno dei suoi aspetti più affascinanti. Proprio come tutti gli altri pacchetti per disegnare, questo può catturare una sezione dello schermo per usarla come pennello. Una volta catturata, si può effettuare ogni modificazione del pennello, come rotazioni e flip-

ping Photon Paint 2.0 trascende queste più semplici funzioni permettendo ai pennelli di essere avvolti attorno ad oggetti. Naturalmente, essi appaiono solamente come se fossero avvolti su un oggetto, ma il programma fornisce abbastanza controllo su immaginarie fonti di luce per rendere l'immagine in modo molto realistico. Alcuni pennelli/oggetti sono predefiniti. coni, cubi, tubi e così via. Questi sono facili da usare e costituiscono un buon terreno di addestramento per imparare a regolare l'illuminazione e gli altri fattori che influenzano l'avvolgimento dei pennelli
Una volta che si domini il tracciamento dei pennelli più semplici, molti si immergeranno nel più complesso mondo dell'avvolgimento dei pennelli sugli oggetti definiti dall'utente. Photon Paint offre abbastanza funzioni in questo ambito perché si possa qualificarlo come un semplice, ma efficace, programma di ray tracing.
Essenzialmente, questo significa che il computer è in grado di far rimbalzare i raggi provenienti da fonti di luce immaginarie sugli oggetti creati dall'artista. Il processo avviene all'interno della "mente" del computer, ma il risultato finale viene visualizzato sul monitor.
Se tentate di usare il programma per avvolgere un pennello sulle curve di livello di una mappa, vi accorgerete che una fase del processo vi permette di orientare il vostro pennello secondo una prospettiva immaginaria (anche Deluxe Paint III incorpora una funzione per mettere i pennelli in prospettiva, ovviamente!). Provate con questi parametri (o altri analoghi a questi) quando fate i vostri esperimenti: Pitch = -17, Roll = 0 e Yaw = 34. Ciò produrrà una vista di tre quarti facile da comprendere, quando farete la preview della versione wireframe, generata dal computer, delle vostre curve di livello. Questa è un buon punto di riferimento per ulteriori modificazioni.
Photon Paint 2.0 include anche un potente pacchetto di animazione che è come al solito ben spiegato in uno dei tutorial.

Questo è un grande programma per realizzare il più delicato lavoro in HAM. Ho speso così tanto tempo con esso, che non sono stato in grado di dedicare lo stesso livello di attenzione ad alcuni degli altri pacchetti. Crea dipendenza ed è un affare a qualsiasi prezzo.

I programmi per l'elaborazione delle immagini

I pacchetti per l'elaborazione delle immagini costituiscono un'aggiunta ai programmi per disegnare e rappresentano uno dei più grandi punti di forza della grafica su computer: essi svolgono funzioni che sarebbero considerati una vera e propria magia sui media convenzionali. Con loro, l'artista può realizzare, per esempio, l'equivalente della trasformazione di una pittura a tempera in un olio divisionista (conversione da HAM a modo grafico a 32 colori), oppure aggiungere un profilo in un lavoro grafico già terminato (varie manipolazioni dei contorni), o ricolorare un'immagine attraverso una versione negativa di se stessa (un'opzione spesso chiamata negative).
Colors in Deluxe PhotoLab è uno di questi programmi, ma i due che seguono costituiscono gli standard più apprezzati.
Ci sono notevoli somiglianze tra di loro, così potete comprarne uno solo. Questo tipo di pacchetto è essenziale per sfruttare pienamente Amiga come macchina grafica. Compratene uno e fate esperimenti con esso anche se non siete del tutto sicuri di quello che fa. Incomincerete presto ad apprezzare il potere che è in grado di esercitare.

PIXmate

Questo programma converte uno schermo Amiga da un qualsiasi modo grafico a qualsiasi altro. Inoltre regola e analizza le palette; offre vari effetti per i contorni; può ridimensionare le immagini e ha un insieme di opzioni di elaborazione

Gli effetti di esaltazione dei contorni di PIXmate producono immagini interessanti

delle immagini così esteso che io stesso non ne ho esplorato che un decimo. Uso il programma principalmente per creare maschere per miscelare complessi oggetti digitalizzati con sfondi fortemente dettagliati.
Ecco come lavoro. Carico l'immagine digitalizzata (HAM) e la converto nel modo a 32 colori.
Poi uso il programma per ordinare e ridurre il numero di colori nell'immagine Spostando i colori nella palette, ormai molto più piccola, è possibile generare rapidamente uno schermo positivo o negativo da usarsi come maschera, che può sfruttare la trasparenza del colore 0 di Amiga.
Questa maschera viene poi caricata in un programma per disegnare, in cui le aree trasparenti sono usate per catturare le aree corrispondenti dello schermo, mentre il resto dell'immagine scompare sotto un colore non trasparente.

Butcher

Pur essendo un potente programma di conversione e di manipolazione dei colori, Butcher baratta un po' di potere nell'elaborazione delle immagini con strumenti che permettono al grafico di modificare i dettagli di un'immagine.
Funzioni aggiuntive forniscono stupefacenti effetti di tipo video, che vi permettono di manipolare un'immagine in molti modi creativi.

Il pennello è nelle tue mani

Sebbene ognuno di questi programmi possa servire come veicolo per esplorare una grande quantità del territorio grafico di Amiga, ci sono anche molte altre applicazioni disponibili per scopi artistici specialistici.
Alcune sono dedicate al già citato ray tracing, che permette agli utenti di sviluppare immagini bidimensionali con l'aspetto di oggetti illuminati e ombreggiati tridimensionalmente.
Altro software è dedicato all'animazione, e offre effetti sofisticati ai disegnatori attirati dalla creazione dell'illusione del movimento.
Naturalmente, l'uso di Amiga in una varietà di situazioni video è ben nota ed è disponibile un numero di applicazioni finali che facilitano questo tipo di lavori.
Il computer è anche stato coinvolto dalla rivoluzione del desktop publishing e l'eccellente output PostScript è disponibile in Professional Page della Gold Disk e in un numero crescente di altri programmi.
Il futuro degli artisti di Amiga appare senza confini. Con lo sviluppo di software sempre più potente e vario, nuovi ampi territori si apriranno, mentre quelli già esplorati saranno migliorati e portati ad un più alto livello.
Una volta che voi abbiate imboccato questa strada, non si può dire dove possiate finire. La sola certezza è che il viaggio merita di essere fatto. ▲

Articolo tradotto dal n.4 Ottobre 1989 di Compute!'s Amiga Resource.

Profilo di un artista KEVIN LUDE

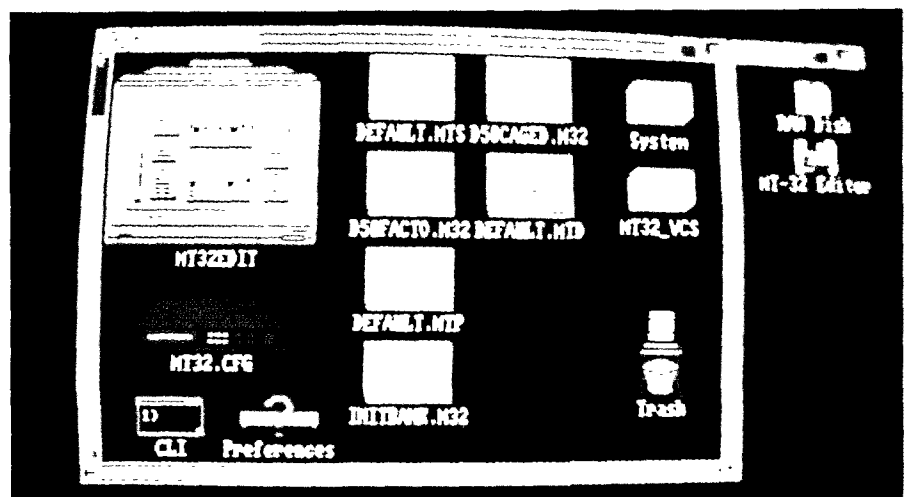
Come pittore ho provato una grande soddisfazione nel sostituire i materiali tradizionali con i programmi per disegnare di Amiga. Un bravo pittore che lavora sulla tela si sforza di simulare la luce, mentre un artista su computer dipinge direttamente con la luce. Amiga offre all'utente l'accesso a una palette di migliaia di colori, precisi strumenti grafici e fantastici effetti speciali. Personalmente, lavoro con questi strumenti per creare nuove mescolanze di stile e di contenuto, e mi sforzo di premere sui limiti presenti nella sensazione visiva.
Le pitture "War Morning" (mattino di guerra) e "Picture Window" (finestra della pittura) sono state prodotte usando Digi-Paint della New-Tek. "War Morning" era la quarta immagine che ho fatto con Amiga. Vinse il primo premio di un concorso, superando altri lavori realizzati con macchine da 10000 dollari (circa 12.500.000 lire). L'immagine vuole essere una sorta di commento sulla sopravvivenza dello spirito umano.
Realizzai lo sfondo di "War Morning" usando il modo Tint con l'opzione Fill. Poi mossi il mouse tutt'intorno, disegnando rapide e casuali figure a forma di otto in toni verdi e blu. Il nero fu aggiunto in alto con ripetuti strati di colore per rendere l'effetto delle colline in lontananza e quello dell'atmosfera. Creai il volto sferico con un cerchio solido; poi usai lo strumento Again di Digi-Paint per le ombre. Per amalgamare il volto con il resto dell'immagine, feci cadere su di esso punte di colore prese da altre parti dell'immagine. Disegnai i capelli con rapidi colpi di uno strumento per le linee e poi disegnai l'aureola con un cerchio colorato.
L'uccello e il recinto furono disegnati a mano libera per dare un tocco più vivo al tutto.
"Picture Window" è una fantasiosa astrazione sulla vista da una finestra di un cottage al mare.
L'immagine ha uno sfondo composto da tonalità di colore realizzate accostando colori primari. La maggior parte del resto dell'immagine fu fatta con figure solide e piatte, ma alcune onde usano la funzione XOR. Aggiunsi le ombre come tinte scure sotto le figure. L'illuminazione frontale e la mancanza di profondità si combinano per conferire all'immagine uno stile che ricorda quello di una vecchia scatola delle ombre cinesi. Se desideraste ricevere due "Hamshows" con questi e altri lavori, un notiziario e un'iscrizione ad Amiga Artist's Club, inviate 5 dollari a Kevin Lude, 4401 Devil's Road, Pemberville, Ohio 43450.

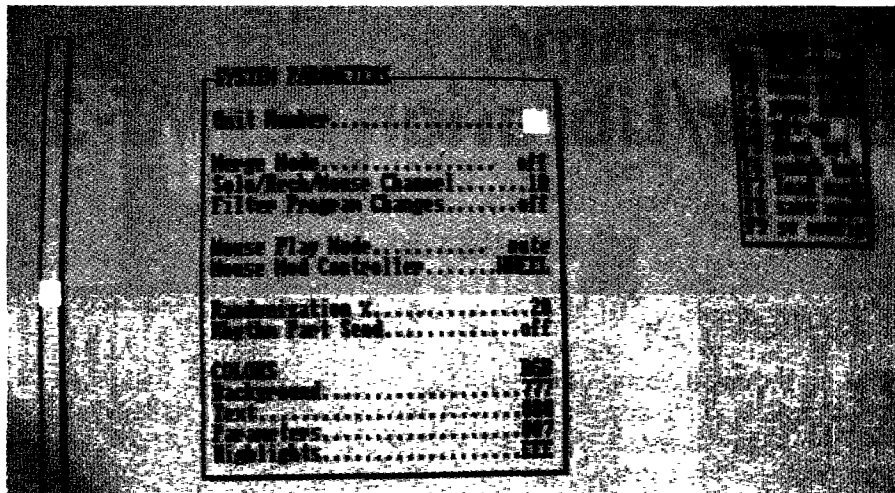
War Morning
Picture Window

Programma Editor/Librarian per MT-32

Un potente software realizzato dall'arcinota Dr.T's per crerare, modificare e archiviare suoni con il diffusissimo expander Roland MT-32.

Aldo & Andrea Laus

Ben tornati al nostro consueto appuntamento con la computer-music; questa volta, come promessò, abbiamo provato per voi una accoppiata molto interessante: l'expander MT-32, della Roland che, grazie al buon rapporto qualità/prezzo, è fra i più "gettonati" strumenti MIDI e la versione Amiga del suo Editor/Librarian, l'MT-32 Editor/Librarian, sviluppato dalla casa Dr.T's per la serie Caged Artist.
Rimandiamo alla scorsa puntata chi ancora avesse le idee poco chiare su cosa è e a cosa serve un programma Editor/Librarian, mentre ora entriamo subito nel vivo del nostro argomento.
Si tratta di un programma molto potente nella gestione dei dati e molto intuitivo nell'utilizzo, tanto da guadagnarsi l'ambito appellativo di "musician friendly"; rappresenta, inoltre, una sicura ancora di salvezza per i musicisti che vorrebbero editare dei suoni sull'MT-32 ma che, dato che la macchina non lo permette direttamente, non lo possono fare. Il programma è contenuto in un solo dischetto ed è dotato di auto-booting (vedi foto 1), ma può essere caricato anche da CLI
La prima operazione che compie è quella di caricare un banco di timbri e alcuni file default per il setup; successivamente si predispone per mandare all'MT-32 una serie di patch, uno per ciascuna delle otto parti gestite dall'expander, se questo non è collegato via MIDI, apparirà un messaggio di errore
Facciamo ora una breve parentesi per capire come avviene la gestione dei timbri, o patch, nell'MT-32.
Normalmente, viene utilizzata la funzione MIDI Program Change per scegliere il suono da assegnare ad una delle varie parti gestite dallo strumento (nel nostro caso otto), ciascuna su un canale MIDI differente.
Il Program Change richiama, quindi, il patch corrispondente dal banco di memoria contenente tutti i patch dell'MT-32 che viene trasferito in un buffer di memoria temporaneo.

La fase di caricamento del programma

2

La prima schermata, per scegliere alcuni parametri relativi alla trasmissione e alla ricezione dei dati

Il patch in questione può quindi essere da voi parzialmente modificato, per quanto riguarda la frequenza, il range del bend, la funzione riverbero e altre piccole cose, ma, purtroppo, non per quello che riguarda la struttura vera e propria del suono.
Con il programma della Dr.T's, si può sopperire a questa mancanza, perché permette di agire direttamente sui quattro oscillatori di ogni suono e, graficamente, sull'inviluppo, oltre che, naturalmente, su tutti gli altri parametri.
I suoni, come sull'MT-32, sono organizzati in banchi singoli di 64 suoni ciascuno; c'è anche il banco dedicato ai timbri di percussione.
Il software, per default, contiene già al suo interno alcuni banchi di suoni, pronti da usare.
Il computer ha bisogno di sapere che timbri sono memorizzati nel banco di memoria dell'MT-32; per fare ciò, crea un banco al suo interno, uguale a quello presente sull'expander; esso lo riflette esattamente e ogni cambiamento effettuato su di esso mediante i comandi Copy, Move, Store, Swap o Load, che vedremo, è riportato anche nel banco dell'MT-32.
Il primo banco di memoria del computer è, per default, questo ma può essere sempre cambiato con il banco secondario.
Il computer, comunque, non tiene in memoria la copia di tutti i banchi dell'MT-32, poiché, visto che questo programma può essere usato in parallelo con un sequencer, si è preferito risparmiare 32K di memoria da dedicare a quest'ultimo; così l'editor lavora a banchi separati: A, B e RITMI.

I parametri di controllo

Il programma si articola in una serie di schermate, facilmente raggiungibili con l'uso dei tasti funzione, ciascuna finalizzata ad una specifica funzione. La prima e più importante di queste si chiama System (vedi foto 2) ed è quella in cui si definiscono i parametri essenziali per la gestione dei suoni.
Questi parametri sono: il numero dell'unità collegata, settato per default a 17, ma modificabile a piacere, facendo attenzione che corrisponda sempre con quello settato sull'MT-32; il modo Merge, che serve per pilotare, durante l'edit, l'MT-32 da una tastiera esterna; il canale MIDI su cui l'intero Editor lavora e su cui comunica con l'expander collegato; il tipo di controllo assegnato al mouse e la percentuale di casualità nella generazione di parametri, durante l'edit dei patch.

Il banco dei timbri

La prima videata di "lavoro" è quella in cui avviene il controllo dei Banchi di suoni (vedi foto 3); è divisa in quattro finestre.
La prima di esse, presente in tutte le schermate, riassume le opzioni possibili e i comandi per raggiungere i vari schermi.
La seconda finestra mostra un elenco delle otto parti e, a ciascuna, associa un patch; in questa finestra si sceglie su quale delle otto parti lavorare; quella scelta verrà automaticamente evidenziata in neretto.
La terza e più "voluminosa" finestra riporta, invece, i 64 patch compresi nel banco scelto; qui si sceglie il suono da assegnare alla parte scelta prima; tale suono viene evidenziato in questa finestra e, contemporaneamente, compare anche in quella precedente, in corrispondenza della parte in questione.
Nella quarta finestra si sceglie, invece, il banco da utilizzare.
In questa schermata sono, inoltre, presenti, nella prima finestra, alcune opzioni per facilitare il compito della gestione dei patch.
Oltre al classico Load e Save, si può inviare, via MIDI, un intero banco all'MT-32, che andrà a rimpiazzare quello precedente; possiamo salvare, all'interno della memoria del computer, un suono da noi editato precedentemente; copiarlo in un'altra posizione o scambiare tra loro due suoni; si può aprire una finestra CLI o stampare l'intero elenco dei suoni; è possibile, infine, uscire da programma con il comando Quit.

L'edit

Con la terza schermata, finalmente, si entra nel vivo dell'edit (foto 4).
L'edit dei timbri è nel nostro caso gestito in due videate complementari; entrambe mostrano tutti i dati comuni, come la frequenza e la forma d'onda.
Mentre la prima videata mostra gli inviluppi TVF e TVA, l'altra mostra solo l'inviluppo della frequenza.
Un'importante caratteristica di questo software è che, nella fase dell'edit, utilizza tre buffer di memoria

(Edit, Compare, Undo), ciascuno contenente otto timbri (uno per ogni parte) Tutti i timbri memorizzati in tali buffer possono essere editati dal programma e rimemorizzati con il comando store.
Il buffer denominato Compare è usato per confrontare due suoni, di solito quello su cui si lavora è l'originale da cui deriva o, se si parte da zero, con un suono simile a quello che si vuole ottenere.
Il buffer Undo, molto utile, memorizza una copia dell'ultima versione del suono che si sta editando, in modo da darci la possibilità, in caso di errori, di ripartire non dall'inizio, ma dall'ultima versione del nostro suono Questo buffer viene messo in funzione con il comando Undo
Ogni suono è formato da quattro algoritmi, ciascuno di essi contiene le informazioni relative alla frequenza, alla forma d'onda, agli inviluppi TVF e TVA; ogni algoritmo può essere escluso dal suono, in modo da poter gestire meglio i rimanenti; è infatti consigliabile, quando si è molto vicini alla versione del suono da noi desiderata, lavorare sui singoli algoritmi separatamente, per fare "un lavoro di fino".
Vediamo ora le varie opzioni disponibili in questa schermata.

Con la funzione Compare/Copy, il timbro corrente viene caricato nel buffer Compare, di cui abbiamo parlato, e mandato al sintetizzatore.
E' possibile, in fase di confronto, copiare da un timbro all'altro singoli parametri, per accelerare l'edit
Altri comandi sono Undo, che, come abbiamo detto, serve a "riparare" ad un errore e Store, con cui un tibro editato viene memorizzato (solo, naturalmente, nella RAM del computer, quindi, per essere sicuri di non perderlo, è meglio farne una copia su disco).

Randomize

Uno dei comandi più interessanti di tutto il programma è certamente Randomize
Esso permette di scegliere il paramentro su cui generare, casualmente, dei valori, dando anche la possibilità di scegliere il fattore di casualità (questo è molto importante, perché ci permette di controllare veramente il lavoro del computer che, in questo caso, diventa un potente aiuto alla creatività del musicista)
Vediamo come si comporta questo comando nei dettagli.
Prima di tutto bisogna crearsi la cosiddetta Maschera: è infatti possibile decidere, volta per volta, quale o quali parametri saranno coinvolti nella generazione casuale di valori, una volta creata una maschera, che, tra l'altro, può essere salvata e successivamente richiamata da disco (evitandoci così ogni volta la fatica di re-impostare la stessa) si passa alla fase di "randomizzazione" vera e propria.
Il fattore di casualità lavora in una scala percentuale da uno a cento e il suo funzionamento risulterà più chiaro con un esempio: se viene scelto un 20%, il programma potrà, all'interno dei parametri selezionati nella maschera, apportare cambiamenti entro un range di valori compreso tra -20 e +20 (il minimo cambiamento può essere + o - 1).
Con il comando Copy/Swap, è possibile copiare alcuni parametri all'interno di un suono (ad esempio, tutti o alcuni parametri da un algoritmo ad un altro).
E' poi possibile eseguire un print out su carta di tutte le informazioni contenute in queste schermate.

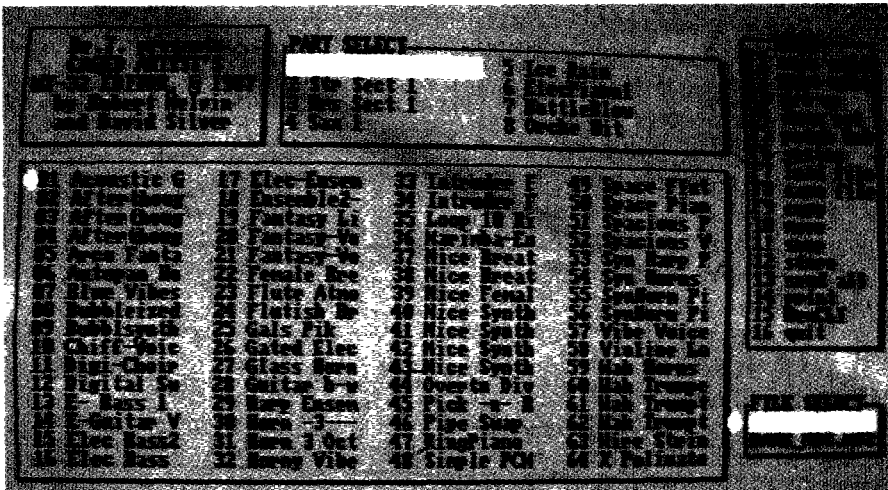
Editing grafico

Ciò che certamente differenzia questo software da molti altri è l'editing grafico, interamente gestito dal mouse, per il controllo dell'inviluppo; è rappresentato in un grafico cartesiano, in cui, all'asse delle ordinate corrisponde il livello di inviluppo e all'asse delle ascisse il tempo. L'intero inviluppo è composto da cinque segmenti che possono assumere, tra loro, ogni angolazione; nei punti di giunzione ci sono dei quadratini, su cui puntare l'indicatore del mouse.
Il livello e il tempo di ciascuno dei primi quattro segmenti determina quello del successivo; il quarto segmento indica di solito il sustain.

Setup

Altra importante schermata è quella detta Setup, che serve a settare, sull'MT-32, alcuni valori per parametri quali: il volume, il tipo di riverbero e il numero di canale MIDI, in modo che corrispondano con quelli impostati in memoria; il tutto per tutte le otto parti, più quelli per la parte relativa ai suoni di batteria. Tali parametri possono essere editati, salvati e caricati da disco; ma non vengono memorizzati in alcun ban-

La schermata in cui vengono assegnati i timbri a ciascuna delle otto parti prima dell'invio all'MT-32

3

co. Nella schermata Patch Bank vengono editati direttamente i timbri presenti nell'MT-32; ci sono 128 timbri di cui 16 visibili contemporaneamente e in ogni momento, anche in questo caso si può sia salvare che caricare un banco.

I timbri ritmici

La schermata che, fin dall'inizio, ha più attirato la nostra attenzione è quella relativa alla gestione e all'edit dei suoni di batteria; chiunque abbia avuto tra le mani un MT-32 non può negare di essere stato colpito dalla fedeltà di tali suoni che, senza dubbio, sono una fra le caratteristiche decisive nella scelta dell'acquisto di questo expander.
Ebbene, se l'MT-32 offre ottimi timbri ritmici, questo programma ne offre un'ottima gestione.
La videata è divisa in tre finestre, la più grande ospita la lista dei timbri, oltre ad informazioni quali il numero di nota assegnato a ciascun timbro, il volume in uscita, il bilanciamento e il riverbero.
Nella seconda finestra si seleziona il range di note da editare; la terza finestra assolve più o meno la stessa funzione: è rappresentata una tastiera in cui vengono segnati i tasti che vengono associati ai vari timbri.
Ogni suono di batteria può essere suonato sia tramite una tastiera esterna, collegata via MIDI (dopo aver assegnato al suono in questione un tasto sulla tastiera), oppure direttamente con il mouse come per i suoni normali.
Anche in questa schermata è possibile salvare e caricare singoli banchi di ritmi; il nome del banco in memoria viene mostrato in alto, sopra la prima finestra.
Il programma permette di utilizzare la funzione MIDI MERGE che combina dati MIDI esterni con quelli di sistema esclusivo generati dal programma.
Quando la funzione Merge è settata su un qualunque valore, che non sia zero, le informazioni che arrivano all'IN dell'interfaccia MIDI sono reincanalate identiche verso l'OUT e mischiate con i dati del sistema esclusivo che escono da questo programma.

Novità Roland

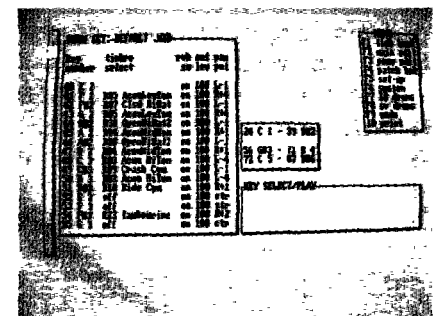
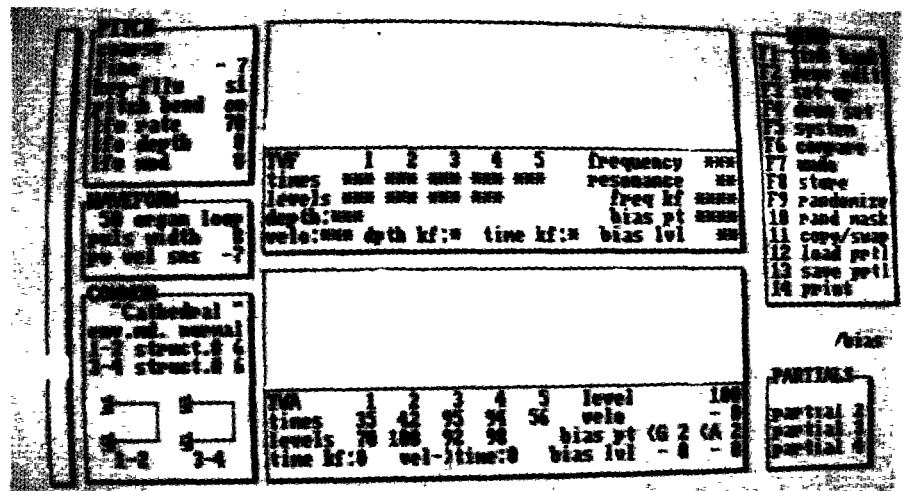
E' di questi giorni l'uscita di due moduli expander Roland per Computer Music che sono compatibili con il programma Editor per l'MT-32 Il primo di essi, CM-32L, è un modulo MIDI che utilizza la stessa tecnica di sintesi dei suoni dell'MT-32 (Lineare Aritmetica); esso contiene, inoltre, una capacità di memoria per effetti PCM pari al doppio di quella presente sull'MT-32, per includere numerosi suoni ed effetti, molto realistici, come il tuono o le onde. Il secondo expander, il cui nome è CM-64, incorpora una vasta gamma di suoni campionati PCM, come il CM-32L, e molti altri suoni in sintesi analogica.
E' equipaggiato di una slot per schede che accetta anche quelle del famoso U-110.
Questa unità permette una polifonia di 63 note; ha una capacità multitimbrica di 15 voci e contiene alcuni effetti digitali

Per concludere

A questo punto la conclusione è ovvia: se il programma della Dr.T's si presentava molto appetibile, viste le sue caratteristiche, per chi dispone dell'MT-32, ora, dopo l'uscita di questi nuovi expander, il suo acquisto diventa di interesse ancora maggiore.
Il programma è corredato, per il momento di manuale in inglese, ma la ditta Soundware, importatore ufficiale dei prodotti Dr.T's, che ringraziamo per averci fornito il software per la recensione, comunica che è in preparazione la versione italiana di tale manuale.

Per ulteriori informazioni, vi consigliamo di rivolgervi telefonicamente alla Hot-Line, che la Soundware ha organizzato per le risposte ai vostri quesiti, al numero 0332/222052. ▲

La pagina di edit e di gestione dei timbri ritmici

L'edit dei suoni Si noti al centro l'editing grafico dell'inviluppo

5

4

Politica commerciale...

Nuove curiosità dal meraviglioso mondo Amiga

Sebastiano Vigna

Qualche giorno fa sono venuto a conoscenza di una notizia che mi ha agghiacciato. la REV6.0 della motherboard di Amiga 2000 (quella con un Mega di Chip RAM), che è stata in vendita per un lungo periodo tempo, è terribilmente bucata! Ci possono essere dei picchi di rumore indesiderati in fase di power-up che impediscono all'autoconfig di funzionare correttamente. I sintomi più visibili sono Guru immotivati all'accensione e non attivazione di schede inserite negli slot. Una volta partito il tutto, comunque, il funzionamento dovrebbe essere corretto.
Quello che potrà stupirvi è che contrariamente alla Commodore-Amiga americana, che rimpiazza in garanzia le 6 0 circolanti con le 6.2, in Europa il comportamento scelto è stato di "stonewalling", vale a dire: "Il problema non esiste, per cui di che vi preoccupate?". L'unico consiglio che posso darvi è se dovete comprare un Amiga, controllate che la REV sia almeno la 6.2 Se avete già la REV6.0 (come me) o la REV6 1 (meglio) cercate di non preoccuparvi troppo, a meno che non pianifichiate di comprare tre o più schede di espansione. Mi hanno detto (ma non essendo un esperto non posso confermarlo) che la sostituzione del 68000 fornito dalla Commodore con uno più veloce, possa attenuare, se non eliminare, i problemi della REV6.x. Sembra essere l'ennesimo scherzo della nostra amata/odiata casa produttrice.
E a proposito di scherzi, ce n'è uno veramente non male da citare nel mio ultimo ordine presso la Creative Computers (la ditta di cui ho parlato l'ultima volta, e che sta aprendo una filiale in Europa), su cinque programmi quattro avevano numeri di versioni fuori commercio da più di sei mesi (sic!), ad esempio XCopy 1.0 o TxEd 2.01. Ho dovuto quindi diramare immediatamente un pacco di lettere ai rispettivi produttori per richiedere nuove versioni, che, evidentemente, non solo dovrò pagare, ma anche aspettare. La cosa, non lo nascondo, mi ha piuttosto adirato, ma, come sempre, questo non influisce minimamente sul fatto in sé Non voglio con questi avvertimenti sconsigliarvi di ordinare software (ben lungi!) ma, più che altro, prepararvi a sorprese sgradite, tipo essere considerati "pattumiere" ove gettare le rimanenze di magazzino, tanto da 10 000 chilometri di distanza non è così facile far sentire le proprie proteste.
Tra i pacchetti su cui ho messo le mani c'è SuperBack, un programma di backup relativamente poco noto in Italia, che presenta una delle interfacce utenti più belle e funzionali che mi sia capitato di vedere su Amiga. Gadget "tridimensionali", shortcut da tastiera per tutte le operazioni, una visualizzazione intuitiva molto chiara della struttura ad albero dell'hard disk, una totale affidabilità e una velocità di tutto rispetto. L'autore, Mike Sinz, attualmente lavora per la Commodore, ma continua (credo) a sviluppare indipendentemente progetti come SuperBack. In senso inverso, vorrei invece riparare e dire qualcosa di male a proposito di GOMF, programma sul quale ho detto ogni lode possibile in un recente articolo. Bene, sembra che sia stato un po' imprudente (capita).

(segue a pag. 29)

PCopy 2.0 & Mostra 1.01

Sebastiano Vigna

Tra i copiatori di public domain disponibili per Amiga, fino ad oggi troneggiava, pressoché irraggiungibile, TurboBackup.
Più veloce dei migliori copiatori commerciali, sebbene non funzionante con software protetto (che in ogni caso va boicottato, giusto?), rappresentava lo stato dell'arte in fatto di duplicazione. Adesso, però, il suo posto è insidiato da un altro bellissimo programma, proveniente dall'Olanda, il cui nome è PCopy
Stando all'autore, Dirk Reisig, la motivazione che lo ha spinto a scrivere il programma è stata la necessità di un copiatore capace di maneggiare agelvolmente situazioni "intasate", del tipo: sto compilando il mio programma C mentre con un amico scrivo un articolo, parlo al telefono, sto scaricando un file via modem e devo fare un po' di copie· chi può stare dietro ad un pasticcio simile ?
PCopy in questo caso ci viene in aiuto con alcune "feature" assolutamente uniche come la history dei dischi copiati fino a quel momento e l'auto-start all'inserimento del disco destinazione. Inoltre, fatto essenziale, il consumo di tempo macchina da parte del programma è veramente irrisorio, cosa ancor più eclatante se si mette in conto che i tempi di copiatura sono di 104/70 secondi rispettivamente con o senza verifica, ovvero ai livelli dei più veloci copiatori disponibili. E' addirittura possibile permettere ai programmi di accedere ad un disco mentre lo si sta copiando, anche se questa operazione può allungare sensibilmente i tempi
Una caratteristica un po' strana del programma è di aprire ben cinque finestre. una per indicare a che punto è la copia, una per i vari gadget di controllo, due per i nomi dei dischi in corso di copiatura e una per la history dei file copiati fino a quel momento.
I comandi sono molto semplici, ma c'è tutto il necessario. go e stop, verify on/off, più qualche opzione per l'autostart.
Per quanto riguarda i difetti, si può parlare solo di mancanza di flessibilità PCopy risolve un problema, e lo risolve bene, ma non risolve tutti i problemi, ad esempio, non è possibile avere più di un target, quindi se avete tre o quattro drive e volete fare copie multiple, dovrete utilizzare TurboBackup o XCopy. Quel che non ho potuto provare è il comportamento di PCopy nei riguardi di un acceleratore 68020 o 68030.
Come ben saprete, XCopy (almeno nella versione 2.0) è pesantemente basato su temporizzazioni "hard" eseguite mediante cicli a vuoto, perciò il tentativo di utilizzarlo su una macchina accelerata potrebbe darvi l'idea di un copiatore che in 30 secondi vi duplica un disco... ma il risultato è solo un supporto magnetico illeggibile TurboBackup, invece, utilizza sistemi più intelligenti (ovvero i timer di sistema), il che lo rende candidato ideale per l'uso su 2500/3000. PCopy "non consuma" tempo macchina, per cui probabilmente si conforma alle buone abitudini di TurboBackup, ma non avendo provato non posso giurarlo.

Mostra 1.01

Vi starete chiedendo che razza di persona può avere chiamato il proprio programma "Mostra" quando tutti, italiani inclusi, danno nomi anglofoni al frutto del loro lavoro. Dato che il chiamato in causa, nella fattispecie, è il sottoscritto, non mi resta che spiegare che il nome non l'ho inventato io !
Originariamente avevo scritto un programma chiamato Show 2 0, che aveva riscosso un discreto successo Ringalluzzito dall'aver ricevuto un certo numero di commenti entusiastici, mi misi a scrivere una versione molto potenziata, ma strada facendo qualcuno mi fece notare che non era poi così facile distinguere Show dagli altri viewer solo in base al nome
A questo punto chiesi di suggerire un nome, e una vera e propria pioggia di proposte (da "SuperShow" a "Isle of View") mi cadde addosso. Nella bolgia generale qualcuno chiese come si diceva "Show" in italiano, e un americano acculturato fornì la traduzione· la parola "Mostra" piacque subito, e infatti, indubbiamente, ora il programma si riconosce Vi starete però anche chiedendo chi mai ha bisogno dell'ennesimo IFF viewer, quando di programmi Show il mondo è pieno
Bene, la mia personale scoperta, dopo una ricerca intensa fatta circa un anno fa per ragioni di lavoro (fare le foto per Bit) fu che non c'erano in giro viewer capaci di mostrare ogni tipo di immagine Se prendevano il PAL, non gli andava bene il mio schermo da 704 pixel, se mostravano l'HalfBrite non mostravano lo SHAM eccetera eccetera. Da qui la necessità di un viewer veramente universale e flessibile.
Mostra è un programma senza compromessi. Vi mostrerà qualunque FORM ILBM presente in un file

IFF che Amiga possa visualizzare. Ad esempio, anche immagini contenute in un file di excellence !
Inoltre, vi permetterà di "scrollare" con i tasti cursore immagini molto più grandi del vostro schermo scompattandole in tempo reale, e non tutte in una volta (naturalmente, occorrerà una quantità di memoria mostruosa!) L'algoritmo di scompattamento è scritto in un Assembler estremamente ottimizzato, e quindi è quanto di più veloce si possa ottenere (su una macchina con 68020 diventa veramente incredibile). Un altro aspetto che ho voluto curare particolarmente sono le opzioni (numerosissime) e i wildcard. Le prime vanno da questioni puramente estetiche (si può, ad esempio, far rimanere uno sfondo nero tra un'immagine e l'altra) a opzioni grafiche (forzare le risoluzioni). Per i secondi, ho scelto il formato che mi sembra più flessibile, ovvero un numero arbitrario di nomi ciascuno dei quali può a sua volta essere una wildcard. "M dh0:* dh1:pics/*", ad esempio, è una sintassi valida. Dato che Mostra usa ARP, i pattern sono più estesi di quelli standard AmigaDOS. Ci sono anche varie potenzialità mirate alla generazione di piccoli slideshow, come un comando di repeat, un'opzione di double buffering e il cambio d'immagine dopo un numero di secondi prefissato.
Per quanto riguarda la documentazione, ci sono due versioni, una in ASCII e una in formato TeX, in modo da rendere disponibile un manuale "professionale" a chi ha accesso ad un sistema di stampa basato sul programma di Knuth.
Mostra è stato duramente beta-testato da diverse persone, e si è rivelato di una stabilità assoluta. Sono comunque interessato a segnalazioni di buchi, modifiche, miglioramenti eccetera.
Il programma, in teoria, sarebbe shareware, con un contributo suggerito di $20, ma non spero certo di diventare ricco in questo modo, e specialmente qui in Italia... ▲

Note della redazione: Mostra è uno dei pochi programmi italiani accettati con entusiasmo dalla comunità -Amiga americana: ci congratuliamo vivamente con Sebastiano per l'ottimo prodotto.

(segue da pag. 27)

Politica commerciale...

In particolare, provate ad usare CED II insieme a GOMF: quando appare il file requester, ecco che la finestrella anti-Guru si anima per dirci che la memoria in pagina zero è stata sporcata... solo che è stato proprio GOMF a farlo !
Il problema risiede in un buco che si rivela solo quando il programma cerca di tracciare una chiamata ad OpenWindow() e la chiamata fallisce. In tal caso, GOMF scrive dei dati in pagina zero e crede (erroneamente) che sia stato qualcun'altro.
Questo e altri problemini analoghi mi hanno convinto, per ora, ad eliminare GOMF dalla mia startup-sequence.
Al suo posto ho inserito un programmino liberamente distribuibile di Carolyn Schnepper, SRT, che intercetta le chiamate ad AutoRequest(), bloccando in particolare quelle del tipo "Software error-task held". In questo modo un buon numero di Guru può essere comunque rimandato a data da destinarsi, ma ricordatevi sempre che non potete sapere cosa un programma ha fatto prima di piantarsi.
L'Exec di Amiga ha un'importante caratteristica: è efficientissimo nel task switching (ad esempio, è molto più veloce di Unix), ma questa sua efficienza si paga ad un prezzo molto caro: non c'è alcuna forma di protezione del codice o dei dati, e cioè qualunque programma può sporcare la memoria che qualcun'altro sta usando.
Il senso di sicurezza che GOMF o SRT danno è dunque in realtà infido, perché proprio nel momento in cui salvate il vostro lavoro potreste, in realtà, stare cancellando l'unica copia decente in vostro possesso. A proposito di questo, Bill Hawes ha detto: "Se pensi di aver bisogno di GOMF, allora stai comprando il programma sbagliato". E il commento mi sembra perfettamente azzeccato.
Quello dell'affidabilità è comunque un problema grave, che dovrà prima o poi essere affrontato. In particolare, è probabile che quando sarà presente sul mercato un numero sufficiente di Amiga dotati di MMU, la Commodore penserà a rendere disponibile, come sotto Unix, una qualche forma di protezione della memoria.
Questo non scagiona comunque le case produttrici che mettono in giro programmi bucati (e non capita solo su Amiga: basta vedere la storia del dBase IV); essi diventano vere e proprie "mine vaganti" che sul più bello, e senza preavviso, distruggono il nostro lavoro.
Da un punto di vista legale non c'è nulla da eccepire, in quanto il contratto di vendita in generale scarica il produttore del software da qualunque responsabilità riguardante la perdita di dati, ma da un punto di vista morale non c'è che da boicottare, per quanto possibile, le vendite del programma stesso
Vogliamo o non vogliamo che Amiga conquisti il suo posto nel mondo dei PC? ▲

3-DEMON

Un programma, veloce e facile da usare, per la creazione di oggetti in tre dimensioni.

Renato Acciardi

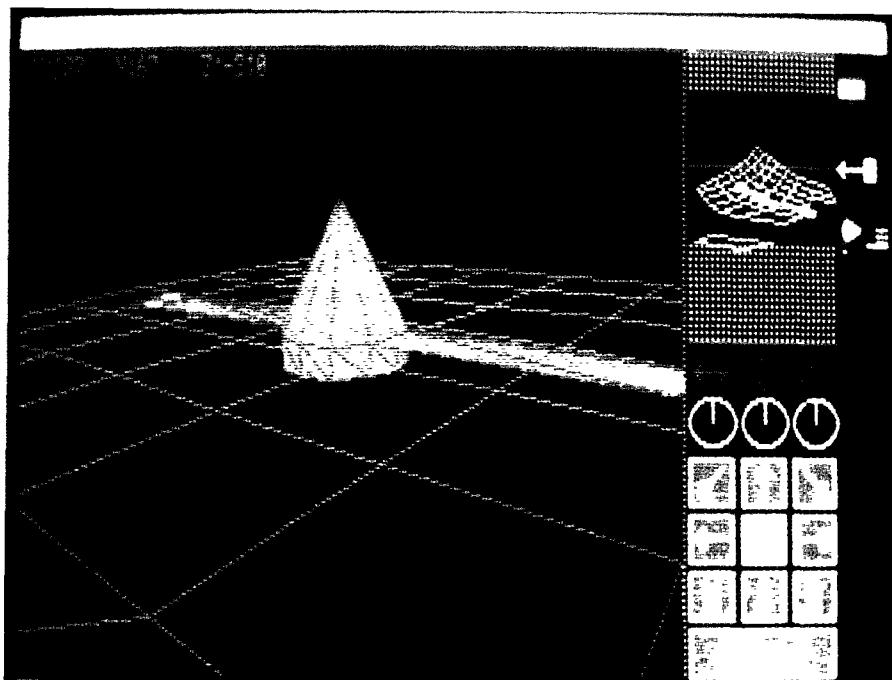
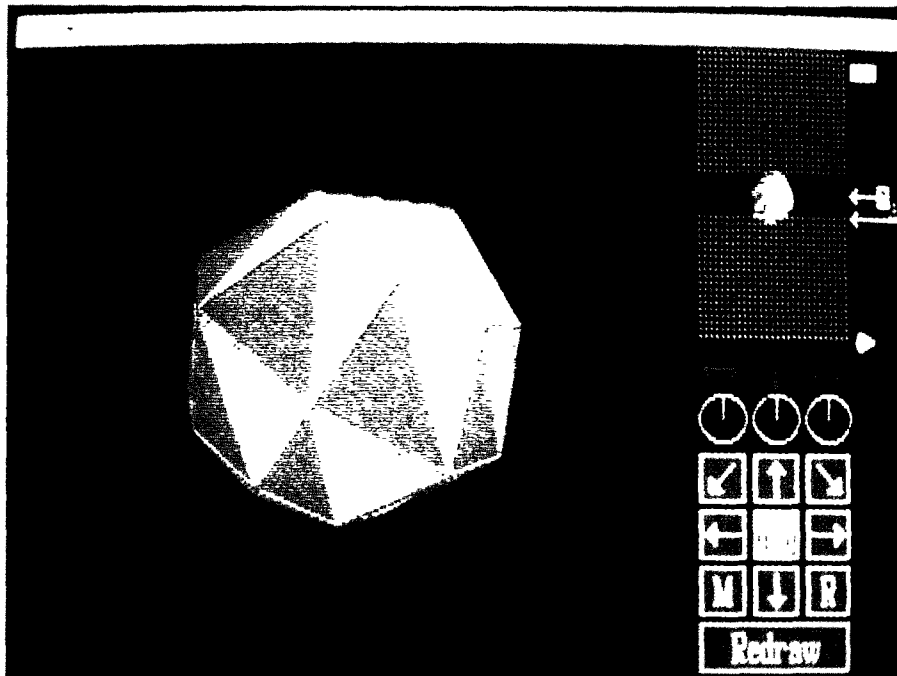
Una parte della computer grafica riguarda l'elaborazione e la visualizzazione di immagini nella maniera più realistica possibile. Questo grazie a giochi di colori, ombre e rifrazioni che rendono le immagini, rappresentate bidimensionalmente sul monitor, più realistiche dal punto di vista della profondità.

Le proprietà grafiche di Amiga hanno permesso la realizzazione di software capace non solo di questo, ma anche di animazioni, ovvero di movimenti predefiniti degli oggetti sullo schermo.

Questi programmi, però, a volte sono carenti per quanto riguarda la creazione delle forme tridimensionali da animare o solamente da rappresentare, nel senso che non permettono, durante la fase di assemblaggio delle varie facce, una completa manipolazione dell'oggetto impedendo così all'utente di controllare completamente il proprio lavoro e, quindi, di portare ottimamente a termine la realizzazione.

Così sono nati dei programmi specifici che agevolano molto l'utilizzatore nella creazione degli oggetti lasciandogli la possibilità di ruotarli, spostarli, ingrandirli e modificarli a suo piacimento anche se non riescono a dare l'idea di quanto questi potranno, effettivamente, risultare realistici una volta colorate e ombrate le facce.

Un programma che è riuscito a limitare questo problema permettendoci di lavorare, in maniera semplice e veloce, sull'oggetto, raffigurandolo con diversi metodi, è 3-Demon.

Nella confezione troviamo, oltre al

Con 3-Demon realizzare oggetti tridimensionali è molto semplice

disco contenente il programma, il manuale (in inglese) e una cartolina indirizzata alla Mimetics Corporation che permette di ricevere informazioni riguardo le novità di questa software house.
Lanciato il programma viene chiesta una parola, diversa ad ogni avviamento, che si può trovare sul manuale alla pagina x paragrafo y. Questa è l'unica protezione del pacchetto che dovrebbe assicurare una minore probabilità di copia illegale del prodotto.

Il Programma

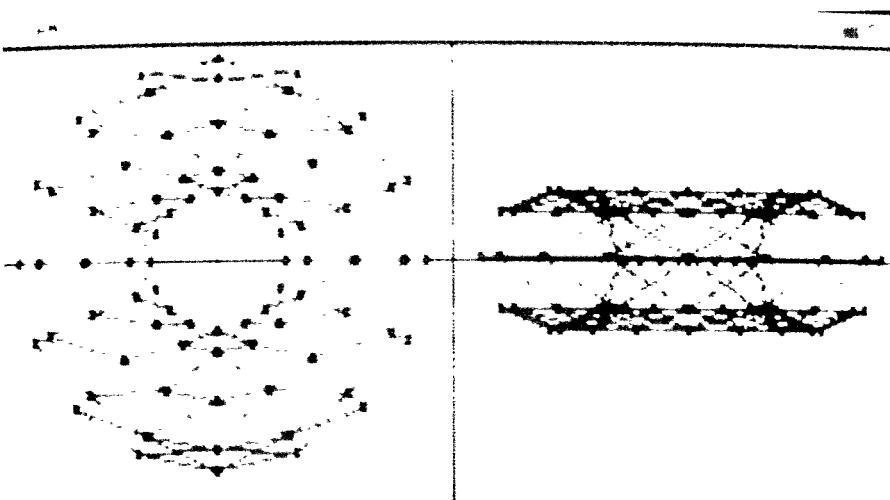
Realizzato dalla Mimetics Corporation, 3-Demon è un creatore di oggetti che possono essere salvati su disco, oltre ovviamente nello standard IFF, in un formato compatibile con i migliori prodotti in circolazione capaci di elaborare immagini tridimensionali o animazioni, ovvero: Videoscape 3D, Silver e Turbo Silver, Sculpt, Gossett Graphics e Forms in Flight.
3-Demon permette di realizzare facilmente oggetti tridimensionali e di visualizzarne le facce con colori diversi, con sfumature di grigio oppure in modo wire frame (ovvero tracciando solo lo scheletro della figura).
Il programma si basa su triangoli per la creazione delle varie facce componenti l'oggetto, il perché è spiegato sul manuale con un semplice esempio: ammettiamo di avere un foglio sul tavolo che rappresenta un poligono (infatti, ha i quattro vertici giacenti sullo stesso piano); se alzassimo un angolo del foglio, quest'ultimo si piegherebbe e avremmo uno strano poligono intrecciato; ma, scomponendo il foglio rettangolare in due triangoli, possiamo muovere qualsiasi spigolo e avremo sempre dei triangoli piani.
Quindi, realizzando gli oggetti mediante triangoli, non si avranno problemi nel muovere un vertice qualsiasi.
Lo schermo è suddiviso in una parte detta Main View e una detta Depth View. La prima occupa i tre quarti dello schermo e permette di lavorare sulle coordinate x e y; per quanto riguarda la profondità si utilizza la Depth View, in alto a destra sullo schermo, che rappresenta l'oggetto sulle coordinate x e z.
Inoltre, vi sono due cursori: uno rappresentante un occhio, l'altro lo schermo che, facendoli avvicinare o allontanare, permettono di variare rispettivamente la prospettiva e la distanza dell'osservatore dall'oggetto rappresentato.
Ovviamente è possibile far ruotare la Main View permettendoci così di lavorare su altre facce.
Un altro tipo di veduta è la Flat Views, ovvero l'oggetto viene rappresentato su due piani, uno di coordinate xy l'altro xz, in modo wire frame, su di una griglia.
Si è detto che il programma si basa su triangoli per la creazione di oggetti.
Per rendere più facile la formazione della sagoma, sono disponibili alcuni modi di aggiungere facce all'oggetto.
Uno di questi è Triangles che ci permette di creare un triangolo semplicemente cliccando sullo schermo i tre punti che ne rappresentano i vertici; Polygons, invece, suddivide direttamente in triangoli un poligono creato unendo i vari angoli.
Un'altra opzione è Lines che traccia una linea selezionando due punti sullo schermo.
Poi vi è Spun la quale, creata una sagoma, la fa girare attorno ad una asse, passante per il primo e l'ultimo punto tracciato, creando un oggetto con un numero di facce che viene richiesto tramite una finestra. Penultima è Sliced che dà spessore ad una sagoma definita. Infine, Back Faces crea un triangolo non rivolto verso l'operatore: questo è molto utile in quanto le facce opposte a quelle rivolte verso l'utente non sono visibili mentre, con questa opzione, si creano delle facce che si possono vedere sia frontalmente che posteriormente.
Molto utili sono i Popup Menu che appaiono, durante l'utilizzo delle funzioni Spun e Sliced, semplicemente cliccando su uno dei punti già creati della sagoma. In particolare le voci dei due menu sono rispettivamente: muovere un punto, far ruotare l'oggetto o interrompere

l'operazione e, nel secondo menu, aggiungere una "fetta", muovere un punto o abbandonare.
Per la selezione dei punti è molto utile il Grab Range ovvero l'intervallo di selezione.
Questo si trova nella Depth View ed è rappresentato da una punteggiatura che sovrasta gli oggetti. Ciò significa che i vertici sotto la punteggiatura non sono selezionati, mentre quelli che si trovano tra le due aree di punti bianchi possono essere modificati. Per spostare l'area di selezione, vi è un cursore apposito (il Grab Range Slider) che la definisce simmetrica rispetto al cursore indicante la profondità (il Distance Slider).
Per ogni faccia è possibile definirne il colore, la quantità di luce emessa e riflessa, la metallicità e la trasparenza.
Naturalmente c'è la possibilità di trasformare l'oggetto: in particolare lo si può muovere, ingrandire o ridurre e ruotare.
E' possibile stabilire una gerarchia tra gli oggetti ovvero definire più oggetti "figli" di un altro: se viene mosso il "padre" anche i figli vengono spostati, ma se viene spostato un "figlio" questo non influenza il resto della "famiglia".

Conclusioni

Dobbiamo ammettere che la Mimetics Corporation ha centrato il suo obiettivo, completando quella parte della grafica che tratta un argomento molto interessante quale è la rappresentazione tridimensionale degli oggetti, realizzando un programma specifico per modificare forme in tre dimensioni. 3-Demon è molto utile, anche se un po' costoso (Lit. 165.000), soprattutto se affiancato ad un programma come Sculpt, che è poco versatile per quanto riguarda la modifica degli oggetti, o a Videoscape 3D il cui editor, separato dal programma vero e proprio, non è del tutto maneggevole. ▲

3-Demon è in vendita presso:
Pix Computer
Via F.D'Ovidio 6c
00137 Roma
Tel.06/8293507

3-Demon si basa su triangoli per la creazione dell'oggetto

AMIGA MAGAZINE

AMIGA Next...

...sul prossimo numero

▲ **Le favolose pagine di Transactor**
Ovvero le pagine del programmatore

▲ **Pacific Motion**
Gli studi di design di Hollywood con la grafica di Amiga

▲ **Virus Protection Toolbox**
Il pacchetto anti-virus più completo

▲ **Bridgeboard**
Una prova pratica della bridgebord Commodore

▲ **Didattica**
Dal 6502 al 68000

▲ **Superbase Professional 3.0**
Un sistema per la gestione di database completamente relazionare di massima qualità

▲ **M**
Un programma musicale semplice da usare ma molto potente

Appuntamento a Settembre in edicola !!

ON DISK è una rubrica mensile di quattro pagine che possono anche essere staccate e conservate, in queste pagine sono descritte tutte le informazioni dei programmi inclusi nel disco, complete di istruzioni, trucchi ecc... In questo spazio troveranno posto giochi, utility e tutto ciò che può fare Amiga.

• Games

Pitch

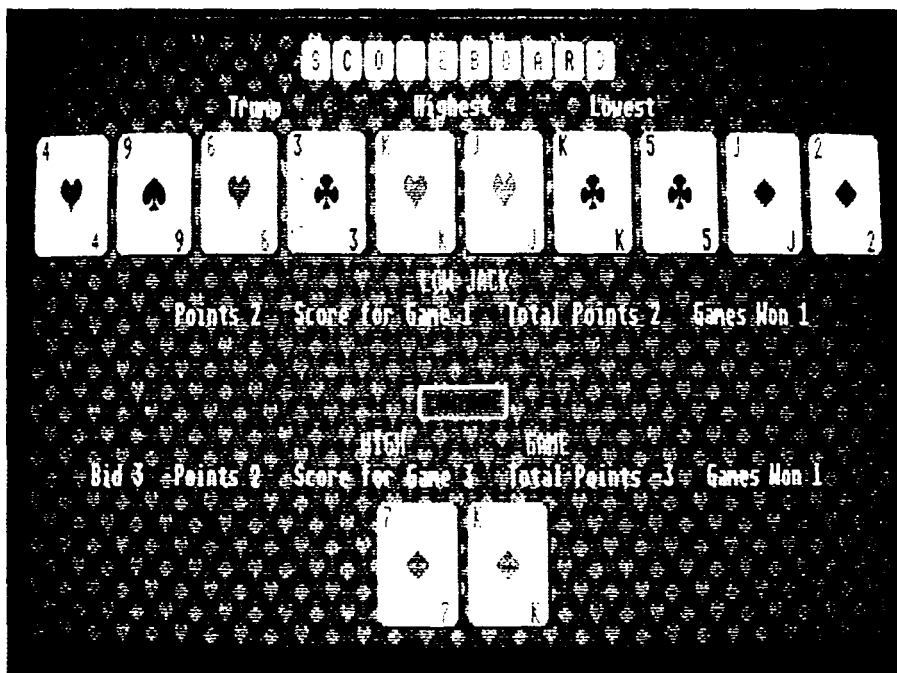
Pitch è un gioco di carte per due giocatori e, nella versione per Amiga, si gioca contro il computer.
Premesso che le regole che determinano il punteggio e danno la precedenza ad uno o all'altro giocatore possono essere meglio comprese giocando, passiamo alla descrizione del gioco.
Inizialmente si decide chi dà le carte, estraendo due carte a caso: chi estrae la carta più alta dà le carte per la prima mano, e successivamente i due giocatori si alternano seguendo la regola del "bid" (approfondita in seguito).
All'inizio della mano ogni giocatore ha una serie di sei carte (prese da un mazzo standard da poker senza jolly, con 13 carte per seme), chi dà le carte decide il proprio "bid", cioè il punteggio che conta di totalizzare in quella mano.
Se l'altro giocatore eguaglia o supera il bid inserito dal primo, diventa mazziere.
Scegliendo un "bid" di quattro ci si assicura di diventare mazziere, ma si corre il rischio di perdere la mano, come si vedrà nella totalizzazione dei punti.
Ogni giocatore può al massimo totalizzare quattro punti in una mano, che si riferiscono rispettivamente alle seguenti situazioni:

a) Possesso della carta più alta del seme iniziale ("Trump")
b) Possesso della carta più bassa del seme iniziale
c) Possesso del Jack del seme iniziale
d) Totalizzazione del più alto punteggio ottenuto sommando nell'ordine:

4 punti per il possesso dell'asso del seme iniziale
3 punti per il possesso del re del seme inizale
2 punti per il possesso della donna del seme iniziale
10 punti per il possesso del 10 del seme iniziale

Questi punti vengono indicati nello Scoreboard con le quattro scritte:

HIGH	per il punto a
LOW	per il punto b
JACK	per il punto c
GAME	per il punto d

A questo punto, proseguiamo con la

descrizione della partita: dopo aver inserito il bid, il mazziere può scartare la carta che preferisce (cliccando con il mouse sulla carta); il seme di quella carta diventerà il "Trump", cioè il seme iniziale.
Ora l'avversario potrà lanciare la sua carta in risposta, per accaparrarsi le due carte presenti sul tavolo e acquisire il diritto a lanciare per primo la carta, la prossima volta.
E' necessario controbattere alla carta dell'avversario con una carta dello stesso seme ma di valore più alto oppure con una carta del seme iniziale ("Trump") su di una carta di un altro seme.
Quando i due giocatori hanno scartato tutte e sei le carte a loro disposizione, appare lo scoreboard e il computer indica chi ha acquisito ognuno dei quattro punto possibili.
Chi ha effettuato il bid ed era il mazziere all'inizio della mano ed è riuscito a totalizzare un punteggio uguale o maggiore del "bid" che aveva scelto, incrementa il suo punteggio totale di quattro punti, in caso contrario perde quattro punti; l'altro giocatore (cioè chi non era il mazziere) aumenta, invece, il suo punteggio totale dei punti totalizzati, qualunque punteggio abbia totalizzato.
Vince chi per primo raggiunge i sette punti nel punteggio totale.

Boomerang

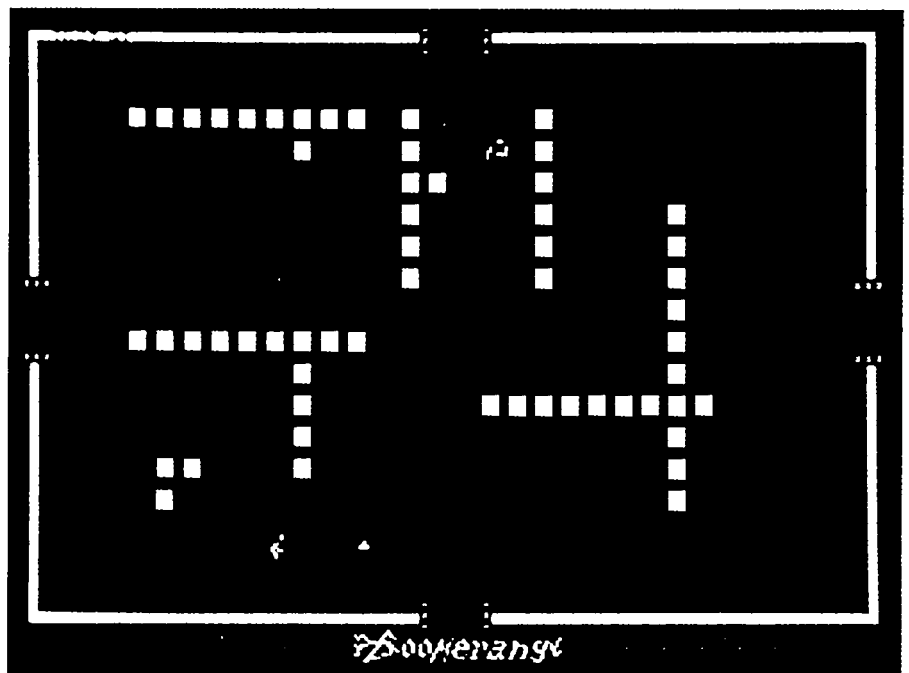
Il Boomerang è stato inventato molto tempo fa da tribù indigene dell'Australia per procurarsi del cibo, ma se utilizzato con precisione può risultare letale anche per l'uomo. E' da qui che prende ispirazione questo gioco. A Boomerang si gioca in due e necessita di due joystick.
La partita si svolge in un'arena che, solitamente, presenta dei muri posti casualmente in modo da ostacolare le possibili traiettorie delle armi dei due giocatori.
Una volta caricato il gioco, è comunque possibile scegliere di giocare in un'arena priva di muri premendo RETURN.
Se invece si preme la barra spazio, verranno creati dei muri all'interno dell'arena, e una nuova configurazione di muri viene creata ad ogni pressione della barra.
Una volta trovata una configurazione di muri soddisfacente, premendo RETURN si può dare il via alla partita.
Il giocatore verde inizia la partita nell'angolo in basso a sinistra, mentre quello viola parte dall'angolo in alto a destra.
Per muoversi è sufficiente utilizzare la leva del joystick e per lanciare il boomerang basta premere il tasto fire. Quando il boomerang è in volo, è possibile deviarlo utilizzando sempre la leva del joystick.
Se il boomerang urta un muro, cade a terra ed è necessario andare a raccoglierlo, ma, con un po' di pratica, è possibile riprenderlo al volo.
L'avversario, colpito dal boomerang, perde una vita e la partita riprende subito dopo che gli omini sono stati riportati alle loro posizioni di partenza.
Ogni giocatore all'inizio della partita ha cinque vite; perde, naturalmente, chi finisce per primo le sue cinque vite.

• Utility

XBoot

XBoot è una semplice ma efficace utility che serve a trasformare il contenuto del bootblock in un file eseguibile, per poterne studiare il comportamento e poterlo analizzare con un normale debugger.
Inizialmente questo programma era stato studiato per scoprire se il bootblock era portatore di virus, ma può rendersi utile anche per trasformare in eseguibile, ad esempio, una delle tante demo che appaiono inserendo nel drive la gran parte dei dischetti presenti in circolazione.
Per ulteriori chiarimenti potete leggere il file documento presente nella directory utility.

AmiCron

Amicron è un'utility che consente di creare un processo in background che richiama a orari prefissati dei

AMOI - Un creatore di isole

Creiamo paesaggi reali con l'aiuto dei frattali

Chris Reynolds e Danny Ross

Danny Ross ha 21 anni ed è programmatore UNIX presso una grande software house, ma lavora saltuariamente a progetti Amiga come freelance. Chris Reynolds ha 23 anni, lavora come programmatore per una grande compagnia di assicurazioni e ha dedicato la sua vita a scoprire perché il primo biscotto di un pacchetto è sempre rotto!
Ciò che segue è la discussione dello sviluppo di un programma per la creazione di paesaggi che è stato presentato nel Marzo del 1989 come tesi di fine d'anno per il corso di laurea in Scienze. E' stato recentemente convertito dal suo formato originale per PC IBM, per poter girare su Amiga ed è stato in parte riprogettato per sfruttare le maggiori possibilità e potenza di elaborazione disponibili.

L'articolo è composto da due parti: la prima è una descrizione delle tecniche, degli algoritmi e dei principi usati nel programma e la seconda è una guida per il suo uso e per creare i vostri paesaggi.
Esse hanno subito solo piccole modifiche per riflettere correttamente la versione per Amiga.

Frattali

L'obiettivo principale di questo progetto stava nella generazione di immagini frattali e pseudo-frattali così come sono state illustrate da Mandelbrot in The Fractal Geometry of Nature, ma indagando i concetti di fondo del suo lavoro, presto compresi che era possibile implementarne, sull'hardware che avevo a disposizione, solo le forme più semplici.

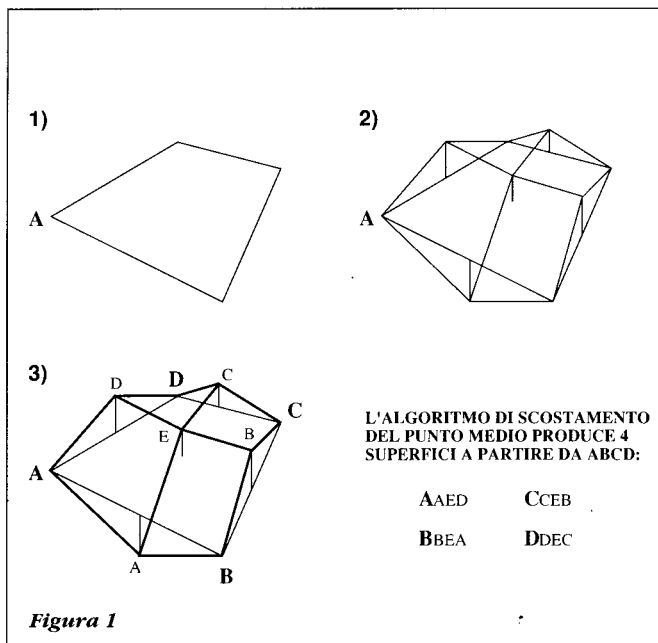
L'algoritmo frattale più semplice è quello detto "scostamento casuale del punto medio". Esso assume che il piano sia composto da una serie di piani più piccoli e che questi piani più piccoli siano composti nel medesimo modo. La generazione casuale di un piano di questo tipo richiede che si prendano i quattro spigoli del piano e si trovi il punto medio di ognuno di essi; poi si muove ciascuno di questi punti medi a una distanza casuale e in una direzione casuale a partire dalla loro posizione originaria. Lo stesso processo è effettuato sul punto centrale del piano.
Quando i punti risultanti vengono connessi, formano un reticolo di quattro piani più piccoli. Il processo è poi ripetuto su ciascuno dei quatro piani e ancora ricorsivamente finché si raggiunge un valore limite prestabilito. Questo concetto è illustrato nella figura 1.

La tecnica è semplice da implementare in qualsiasi linguaggio che consenta delle procedure ricorsive ed è ragionevolmente veloce da eseguire; ci sono molti programmi di public domain che usano questo algoritmo. Tuttavia, il procedimento soffre di alcuni inconvenienti. E' interamente casuale e l'utente non ha alcun controllo sulla forma del paesaggio prodotto; i paesaggi generati, inoltre, sono caratterizzati da un aspetto molto frastagliato e roccioso che, sebbene di grande effetto, simula solamente una piccola sezione dei diversi tipi di terreno formatisi in millenni di erosione. In tutto il lavoro di Mandelbrot non ho mai visto un paesaggio collinoso.

Generare paesaggi reali

Dunque, se le tecniche frattali offrono poche opportunità per la modellazione di paesaggi realistici, perché non usare i dati forniti dalla vita reale e utilizzare autentici dati dell'Ordnance Survey (Istituto Cartografico) per generare i nostri paesaggi?
Ebbene, potremmo farlo; l'organizzazione Ordnance Survey (O.S.) aveva cominciato l'immenso lavoro di riversare su computer le proprie carte topografiche e noi disponevamo di due file della O.S. Il primo era un semplice reticolo delle altitudini di un'area del Galles del Sud e il secondo un tentativo unico di rappresentare tutte le informazioni presenti su una normale carta topografica di tipo cartaceo dell'O.S. che si riferiva a un'area dello Yorkshire. Tuttavia, la mappa dello Yorkshire, da sola, richiedeva qualcosa come 400 Mbyte dello spazio su disco del VAX principale del nostro Politecnico ed era accompagnato da un manuale su disco di 200 pagine: non molti avevano la possibilità di stamparne il contenuto e, in ogni modo, i dati che contenevano erano fissi; produrrebbero ogni volta lo stesso paesaggio e non consentirebbero all'utente di esprimere la propria creatività, cosa che, naturalmente, costituisce la metà dell'interesse.

Si doveva trovare un compromesso. Si decise di permettere

Figura 1

all'utente di specificare le curve di livello (usando una semplice interfaccia guidata dal mouse) e poi di sviluppare il codice per elaborare queste curve, al fine di produrre una proiezione tridimensionale del paesaggio, suggerito dalle curve di livello, con correzioni frattali addizionali, nel tentativo di rendere più realistico il paesaggio. Inoltre, si è definito un algoritmo per fornire al generatore di paesaggi alcune regole per rendere i pendii leggeri e le pianure. Il terreno più basso viene disegnato come sabbia che cambia gradualmente per diventare una fascia di foresta, prima di dissolversi in pendii montani rocciosi.
A parte le funzioni di base, sono state aggiunte opzioni per permettere di salvare e caricare informazioni sulla mappa in tre formati diversi.

1. Come una mappa delle curve di livello. Questa può essere caricata per correzioni o per riprodurre il paesaggio in un momento successivo (forse dopo aver riprogrammato gli algoritmi di proiezione, così da consentire la sperimentazione). Essendo il programma destinato ad Amiga abbiamo deciso di usare il formato IFF per la mappa delle curve di livello. Ciò permette a coloro che possiedono Deluxe Paint, o simili, di modificare le mappe con strumenti molto più potenti di quelli forniti da AMOI.

2. Come un reticolo già elaborato. E' l'immagine spaziale del terreno come risulta dai calcoli effettuati. Salvare un paesaggio in questo formato permette di disegnare i paesaggi senza dover ricalcolare i dati del reticolo a partire dalle curve di livello.

3. Come file in formato IFF dell'immagine finale del paesaggio. Ciò garantisce all'utente una grande flessibilità, in quanto ci permette di scaricare su altre utility la responsabilità di compiti difficili quali la stampa.

Gli strumenti forniti per creare le mappe delle curve di livello sono tutti disponibili sullo schermo principale, dove un'area di disegno piuttosto piccola (256 x 200) appare circondata da una cornice bianca. Le curve di livello vengono tracciate in questo riquadro usando il mouse e l'utente è provvisto di 15 colori (che indicano l'altitudine) per riempire le aree che desidera.
Una descrizione completa dell'interfaccia utente è data più avanti nell'articolo. Dapprima, descriveremo le tecniche usate per trasformare una mappa delle curve di livello nell'immagine finale.

Tecniche di interpolazione

Il cuore del software giace nelle routine che convertono i dati delle curve di livello in un'immagine tridimensionale. Lo scopo principale di queste funzioni è arrotondare l'effetto gradino tra i differenti livelli delle curve.
Il più alto livello di input nell'algoritmo di interpolazione è la mappa delle curve di livello colorata, disegnata dall'utente. L'output della fase di interpolazione è un reticolo di altitudini di 64 x 64. Le righe e le colonne del reticolo si riferiscono alle posizioni x e y della mappa delle curve di livello.

Il primo compito che la routine deve effettuare è l'eliminazione delle curve di livello. Questo perché la routine che gestisce i dati dei livelli usa il colore di un pixel dello schermo per determinare l'altitudine. Dal momento che le curve di livello sono tutte visualizzate nel colore quindici (bianco), l'esistenza delle curve di livello sulla mappa produrrebbe altitudini spurie del valore quindici. Per rimuovere le curve di livello si richiede un semplice algoritmo di analisi dell'immagine.

Fondamentalmente il programma esamina ogni pixel nell'area di disegno e stabilisce se è del colore quindici. Se questo è il caso allora l'algoritmo entra nel modo ricerca e sostituzione. Dapprima guarda un pixel a Nord della posizione corrente e se trova un colore che indica l'altitudine rimpiazzerà il pixel originale con quel colore. Se non viene trovato un colore valido, allora il processo sarà ripetuto per l'Ovest, il Sud e l'Est. Se non si trova un colore valido in nessuna delle quattro direzioni allora il processo viene ripetuto, ma questa volta si esaminerà l'area circostante a una distanza di due pixel. L'incremento della distanza di ricerca viene ripetuto finché non si trova un colore valido.

In pratica, siccome le linee sono raramente più larghe di uno o due pixel, un colore valido viene normalmente trovato nel corso del primo o del secondo giro attraverso i punti cardinali. Un'eccezione si incontra quando sia stata effettuata una grande quantità di correzioni su una particolare area dello schermo. In questo caso, può essere necessario un numero più grande di iterazioni; l'accuratezza di questo algoritmo diminuisce man mano che aumenta la distanza dal punto iniziale.
La fase seguente è per molti aspetti la più importante, in

quanto è in essa che vengono rimossi i gradini fra i dati. Ciò che segue illustra le regole fondamentali della routine.
Ogni punto di un'area con una data altitudine è probabile che si ponga ad un'altitudine compresa tra il punto più basso di questa regione e il punto più basso della regione immediatamente superiore. Ovviamente, questa altitudine non sarà la media delle altitudini delle due regioni, ma sarà invece in proporzione. Detto più semplicemente, l'altezza di un punto in una particolare regione è dettato dalla vicinanza alle regioni adiacenti.

Così, un'ovvia implementazione di questa teoria potrebbe procedere esaminando i punti che confinano con ciascun punto del reticolo, stabilendo l'altitudine di quest'ultimo in base all'altezza dei primi. Per esempio, se un punto di altitudine 5 è circondato da due punti di altitudine 2, tre di altitudine 6 e tre di altitudine 5, allora in quanto ha tre punti adiacenti dal valore 6 e solo due di valore 2, potremmo assumere che esso si pone a tre quinti della distanza fra i valori 5 e 6: per esempio (3/(2+3)). Così stimiamo la sua altezza (5+(3/5)) o 5.6. Se noi ripetiamo questo calcolo per ciascun punto del reticolo dovremmo arrivare a una versione approssimata della mappa.

Ci sono inconvenienti in questo approccio. A meno che le curve di livello non siano state riempite molto bene, accadrà che molti punti avranno punti adiacenti di differenti altitudini e solo quelli che giacciono vicino alle curve di livello beneficeranno della tecnica di approssimazione. Ci sono due ovvie soluzioni al problema.
La prima consisterebbe nell'analizzare la mappa a partire dal punto e fino ad incontrare le aree dall'altitudine più bassa e più alta (in maniera analoga alla tecnica di rimozione dei contorni, descritta in precedenza) e poi usare un calcolo similare per stabilire l'altezza reale del punto. La seconda soluzione sarebbe quella di permette che il punto interessato sia modificato da tutti i punti presenti in una certa area attorno ad esso.

Il nocciolo di questa seconda soluzione è generare un secondo reticolo che abbia, per ciascun punto, la somma di tutte le altezze presenti in un quadrato attorno a quel punto, poi dividere il totale per il numero di punti sommati e usare questo valore medio come nuova altezza del punto. Un problema si genera per i punti vicini ai margini del reticolo, perché non possiamo stabilire il valore dei punti al di fuori del reticolo e, inoltre, se saltiamo quei valori, otterremmo un'altitudine più bassa di quella vera, perché la somma sarà ancora divisa per il numero totale dei punti presenti in un quadrato normale. La soluzione è costruire un array di aggregati i quali contengano, per ciascun punto, il numero di punti che hanno influito su di esso: a questo modo, quando verrà eseguita la fase di calcolo delle medie, il totale sarà diviso dal valore contenuto nell'aggregato e non dal numero totale dei punti in un quadrato standard. Ho provato entrambi questi algoritmi e posso affermare che producono un'interpolazione piuttosto realistica dei dati delle curve di livello. Lo stadio successivo del sistema riguarda la rappresentazione su uno schermo bidimensionale del reticolo dei punti.

"Frattalizzazione", proiezione e ombreggiatura

L'algoritmo di proiezione tridimensionale costituisce un processo abbastanza semplice che mette in scala l'ampiezza dei quadrilateri, che costituiscono l'immagine finale, a seconda della loro distanza dall'osservatore. Cambiare l'intervallo con cui le dimensioni aumentano o diminuiscono influisce sul punto di vista apparente dell'utente. Le prime versioni del programma avevano un punto di vista molto più alto che sembrava essere molto più lontano dal paesaggio. La versione finale usa un punto di vista che dà all'utente l'impressione di essere "sul" terreno.

La prima cosa che il generatore di paesaggi fa, è definire un piano "a livello del mare" di colore blu che è una semplice proiezione dei quattro vertici con elevazione pari a zero. Poi effettua una traversata del reticolo di 64 x 64 altitudini a partire dal fondo e verso la parte anteriore. Per ciascun punto del reticolo (i,j), viene prima formato un piano con i punti (i,j), (i+1,j), (i+1,j+1) e (i,j+1) e poi chiamata la funzione principale di proiezione draw_poly( ). La funzione accetta come parametri le coordinate del reticolo e calcola le altezze per ciascuno dei vertici del piano.

Draw_poly( ) è una routine ricorsiva basata sull'algoritmo di scostamento del punto medio di Mandelbrot, descritto in precedenza. Inizialmente essa proietta due dei vertici del piano, ridefinendoli come coordinate x, y relative ai pixel di Amiga e fa una prova (con un piccolo fattore casuale) per determinare se il piano ha raggiunto il suo limite massimo. Il limite massimo del piano dovrebbe essere, idealmente, quel punto in cui il singolo dettaglio ha già raggiunto i limiti della risoluzione dello schermo e ogni ulteriore esame del piano sarebbe improduttivo. Le prove hanno dimostrato che un piano più piccolo di uno largo 6 pixel e alto 5 non aggiunge più "realta" se sottoposto ad ulteriore "frattalizzazione".

Se questo è vero per il piano, le coordinate vengono passate oltre per la fase di visualizzazione. Altrimenti draw_poly( ) genera uno scostamento in altezza (basato sull'altezza iniziale) per il punto medio della superficie e chiama se stessa quattro volte, una per ciascuna delle superfici appena create. Lo scostamento del punto medio può essere identificato, nel listato, dal prefisso nz.
Quando una superficie viene finalmente accettata, essa passa attraverso due algoritmi. Il primo è un tentativo di incrementare l'efficienza delle routine di proiezione ed è una tecnica conosciuta come "back plane removal" (rimozione dei piani posteriori). Questa semplicemente determina se un piano non è rivolto verso l'utente, selezionando per la visualizzazione solo quei piani che gli stanno di fronte. Sebbene il principio sia semplice, richiede abbastanza matematica.
Per determinare se abbiamo a che fare con un piano

posteriore è necessario calcolare la normale alla superficie del piano e il vettore della normale al piano di visione. Se il prodotto scalare di questi due vettori è negativo allora il piano è un piano posteriore e non deve essere disegnato. Perché questo algoritmo funzioni, è necessario che i vertici di ogni poligono che fa parte della figura, siano ordinati in modo antiorario (assumendo un sistema di coordinate destrorso).

Ombreggiatura delle superfici

Quando gli algoritmi di proiezione hanno deciso che un piano può essere visualizzato, il prossimo passo consiste nel calcolare l'intensità della luce riflessa dalla superficie del piano.
L'intensità dipende da quattro fattori:

Ip intensità della sorgente luminosa
kd coefficiente di riflessione della luce della sorgente luminosa
Ia intensità della luce ambientale
Ka coefficiente di riflessione della luce ambientale

Dal momento che la luce ambientale è costante, qualsiasi sia la rotazione dell'oggetto, non è necessario ricalcolarla ogni volta. La quantità di luce della sorgente che viene riflessa verso l'osservatore, dipende dall'angolo di incidenza tra il vettore della sorgente luminosa e la superficie considerata. L'equazione che esprime l'ombreggiatura è rappresentata da:

I = Ia * Ka + Ip * kd * (L.N)

L.N è il prodotto scalare fra il vettore normalizzato della normale alla superficie e il vettore normalizzato della sorgente luminosa. Il risultato I dell'equazione è l'intensità della luce riflessa dal piano ed è usato, nella fase di visualizzazione, per modificare i colori al fine di rendere l'ombreggiatura del piano. E' necessario definire un valore limite per I, così che rimanga entro i confini della tavola dei colori.

Nella versione 2.7 di AMOI, il vettore della sorgente luminosa (situata in qualche luogo sulla destra del paesaggio) è una costante, definita dai tre valori lx, ly e lz. Se desiderate, potete alterarli per spostare la sorgente luminosa. Si noti che lx, ly e lz formano un vettore normalizzato. Per coloro che non lo sanno, ciò significa che non si può semplicemente andare ad alterare lx, ly e lz a piacimento.
La funzione, chiamata plane_colour32( ) nel codice sorgente, è ragionevolmente incomprensibile, ma se voleste scavare in essa, potreste notare con piacere che, per calcolare la normale a una superficie, si richiede solamente che siano noti due vettori del piano. Questi sono rappresentati dai vettori (x0, y0, z0) - (x1, y1, z1) e (x1, y1, z1) - (x2, y2, z2).

Il risultato utile di tutte queste routine è che, talvolta, la proiezione di un paesaggio stupendo e la combinazione di distese di sabbia, foreste e rocce può farvi arrivare alla vostra lozione abbronzante e al bar Bounty prima di sapere che cosa vi stia capitando.
Naturalmente, la prima cosa che vorrete fare sarà mettere sotto pressione la vostra Xerox 4020 e stampare delle cartoline postali, che riportino il solito compiaciuto messaggio del tipo "Qui il tempo è..., spero che tu stia bene", da inviare ai vostri amici. Bene, per soddisfare proprio queste esigenze, AMOI fornisce un'opzione per salvare file in formato IFF.

Il supporto IFF

Come è stato spiegato nei paragrafi iniziali, la gestione dei file IFF aumenta in maniera consistente la flessibilità di AMOI. Sfortunatamente, questo significa che si deve scrivere moltissimo codice di supporto per fornire le funzioni IFF di lettura e scrittura. Fortunatamente, gran parte del lavoro è stato già fatto per noi da molte persone gentili.
AMOI usa la struttura BitMap di Amiga come base per le sue operazioni di lettura e scrittura, in quanto, entro la struttura BitMap, troviamo quasi tutto ciò che ci interessa sapere sull'immagine in ingresso o in uscita.

Quando si carica o si salva un'immagine del paesaggio, le routine vengono chiamate usando la struttura BitMap dello schermo visualizzato, così le operazioni vengono svolte direttamente sul video. L'informazione sui contorni, tuttavia, viene gestita in maniera un po' diversa, in quanto la mappa delle curve di livello è contenuta entro un reticolo di 256 x 200, su uno schermo di 320 x 256. Molto semplicemente, manteniamo un secondo buffer per la BitMap, che verrà caricato a tempo debito con le informazioni sulle curve di livello, attraverso operazioni del Blitter da e verso la BitMap dello schermo.

Nonostante l'IFF sia esaminato in molte pubblicazioni (specie nell'appendice di Amiga ROM Kernel Reference Manual: Includes And Autodocs, Addison-Wesley,1989), esso occupa una porzione significativa del listato e le sue routine potrebbero benissimo tornare utili, in altri momenti, a quei programmatori che volessero usare file IFF (anche solo per caricare un'interfaccia predisegnata). Per capire il supporto IFF è necessaria una definizione del formato IFF.

IFF e IFF ILBM

Il formato IFF per i file fu progettato dalla Commodore e dalla Electronic Arts, e costituisce il tentativo di fornire uno strumento per immagazzinare i dati in una forma che sia indipendente dalla macchina. Ciò non riguarda solo le immagini, ma anche i suoni, le animazioni e altro ancora. Il formato è diventato rapidamente lo standard per il formato dei file su Amiga e sta rapidamente raccogliendo favori in altri ambienti.
Il formato è basato sul concetto di blocchi (chunk) indipendenti; ogni chunk ha un header (intestazione) standard di otto byte, i quali contengono il nome del chunk (di solito

quattro caratteri alfanumerici come BODY, VHDR, ILBM) e la sua lunghezza. Grazie alla presenza della lunghezza, un programma può saltare un chunk sconosciuto, leggendo (e ignorando) il numero specificato di byte, per arrivare all'inizio del chunk successivo.

Con la maggioranza dei file IFF si può contare sul fatto che il primo chunk sia del tipo FORM. File IFF più complessi possono essere costruiti attorno a tipi CAT e LIST, ma non capita spesso: se volete, potete anche preoccuparvi di loro, ma AMOI è già abbastanza lungo così. Dunque avremo:

4 byte 'FORM' (che identificano questo come file IFF)
4 byte lunghezza del FORM (lunghezza del file)
4 byte 'ILBM' (tipo di file IFF)

Poi segue una serie di chunk, ciascuno dei quali è un elemento indipendente del FORM. Si presuppone che un chunk sia indipendente dalla posizione che ha nel file: per esempio, il chunk BMHD in un file IFF potrebbe venire prima o dopo il chunk BODY, ma in alcuni casi questo può risultare poco pratico.

chunk BMHD
4 byte 'BMHD' (BitMap HeaDer, chunk di intestazione della BitMap)
4 byte lunghezza del chunk
20 byte dati di intestazione della BitMap

Il chunk di intestazione della BitMap offre informazioni fondamentali sul formato dell'immagine, come l'ampiezza orizzontale, quella verticale, il numero di bitplane, la tecnica di compressione usata e così via. Tutti i dettagli si possono vedere nella funzione LoadIFF( ) nel codice sorgente.

chunk CMAP
4 byte 'CMAP' (Colour MAP, chunk della mappa dei colori)
4 byte lunghezza del chunk
n byte informazioni sui colori

CMAP è un chunk a lunghezza variabile che contiene 3 byte per ogni colore dell'immagine. Questi byte rappresentano i componenti Rosso, Verde e Blu dei colori e viene coperta l'intera gamma 0-255.

Nel caso vi stiate chiedendo perché si preveda la gamma 0-255, quando Amiga può visualizzare solo le intensità 0-15 di ciascun colore, dovreste ricordare che l'IFF non è uno standard solo per Amiga.
Può capitare che l'immagine debba essere visualizzata su un sistema che possiede una gamma maggiore o minore di quella prevista sull'attuale macchina ospite, così il formato IFF deve riportarla nella gamma 0-255 [usando solo i 4 bit più alti del byte, N.d.T] perché si possa adattare a qualsiasi altra macchina. A questo modo, nessuna informazione sul colore va perduta.

chunk BODY
4 byte 'BODY' (i dati dell'immagine)
4 byte lunghezza del chunk
n byte di interleaved bitmap (bitmap "interfogliata" o "intercalata")

Come il nome suggerisce, il chunk BODY (corpo, massa) contiene tutti i dati effettivi dell'immagine. Tuttavia l'immagine non è salvata come si potrebbe immediatamente immaginare. Il principio su cui si fonda è che, per ogni linea dell'immagine, esiste una linea corrispondente in ognuno dei bitplane di quell'immagine. E' di qui che proviene l'espressione interleaved bitmap; un file IFF ILBM conterrà, dunque, la prima riga del bitplane 0, la prima del bitplane 1, la prima del bitplane 2 e così via, poi la seconda riga del bitplane 0, la seconda del bitplane 1 e così via. A questo modo viene costruita l'intera immagine. Questo formato è particolarmente adatto al sistema video dell'Amiga.

Compressione ILBM

L'informazione grafica ha generalmente molte pretese per quanto riguarda la memoria. Fortunatamente, è di solito anche molto adatta alle tecniche di compressione, per cui grandi aree di uno stesso colore (per esempio, gli sfondi uniformi di molte immagini) possono essere rappresentate da pochi byte.
Il chunk IFF FORM ILBM definisce una tecnica di compressione chiamata Byte Run encoding (codificazione delle sequenze di byte) che si applica a ogni riga di ogni bitplane, quando vengono "intercalate" fra di loro.
Quando i dati vengono compressi, li si esamina per stabilire se esiste una sequenza di byte ripetuti. Se una sequenza (run) è riconosciuta, la routine di compressione inserisce un byte di controllo che indica la lunghezza della sequenza e lo fa seguire dal byte che deve essere ripetuto. Gli intervalli tra sequenze (quando nessuna serie adatta alla codifica sia stata rinvenuta) sono segnalati da un diverso codice di controllo, che indica in questo caso la lunghezza della "non sequenza". Per distinguere tra un controllo di sequenza e un controllo di "non sequenza", viene alterato il segno del byte:

I valori da 0 a 127 indicano una "non sequenza" lunga da 1 a 128 byte
I valori da -1 a -127 indicano una "sequenza" lunga da 2 a 128 byte

Commenti generali al programma

AMOI fa perno sul familiare ciclo di attesa attraverso un Port per messaggi IDCMP. Quando un'azione dell'utente viene segnalata, il programma esegue un'ampia istruzione switch, per determinare il tipo di input ricevuto. La selezione dei menu è un modo semplice per avviare le funzioni più importanti del programma. La selezione dei gadget della gamma dei colori (definita come un solo grande gadget) e del gadget Draw/Fill è trattata allo stesso modo. Per calcolare quale colore sia stato selezionato, si determina la

posizione del mouse rispetto all'asse delle y, eliminando l'occorrente per 14 gadget (uno per ogni colore).

L'azione successiva è costituita dalla pressione del bottone sinistro del mouse, che funziona da interruttore per il tracciamento del mouse. Se l'utente ha tracciato una linea di contorno al di fuori dell'area per disegnare, ogni disegno viene sospeso finché il puntatore non rientra nel riquadro. Quando il disegno viene eseguito, tracciando una linea dall'ultima posizione conosciuta del mouse fino alla posizione corrente, viene alzato un flag (FLOST) che fermerà il disegno quando il mouse fosse trascinato al di fuori dell'area di disegno (cosa che potenzialmente produrrebbe una linea spuria, metà dentro e metà fuori dal riquadro) e viceversa. Sebbene ciò non sia affatto all'altezza della classe degli algoritmi di clipping di Cohen-Sutherland, è un rattoppo sensibilmente più corto e funziona (ma si stia attenti alla formazione di interruzioni quando si entra o si esce dall'area di disegno: il flusso del colore si infiltrerà attraverso di esse).

Usiamo FastFlood( )

Se avete letto l'articolo "Algoritmo di riempimento veloce" nel numero 6, 1989, p.27 di Transactor per Amiga, saprete tutto sul veloce sostituto della funzione buit-in Flood( ), che è stato ora incorporato in AMOI (ebbene, AMOI fu il motivo ispiratore del suo sviluppo). Il codice di AMOI assume che si stia usando il file fastflood.h e, a causa di ciò, contiene una breve sequenza di codice che controlla se il programma è stato lanciato con una sufficiente quantità di stack.

Come si nota anche nel listato, penso che ci dovrebbe essere un sistema migliore per trovare l'ampiezza dello stack del processo: io non so quale sia.
Il metodo che uso consiste nell'attraversare molte strutture DOS per trovare lo stack di default di questa CLI. Forse ricordate che FastFlood( ) richiede una notevole quantità di stack e così il codice non permette all'utente di proseguire finché non determina la presenza di uno stack di 40000 byte o più. Se usate FastFlood( ) ricordatevi che dovete fare il link con il file oggetto fastpixel.o, ottenuto dal codice di supporto assembler di FastFlood().
Se non avete il codice di FastFlood() (o se, più semplicemente, vi piace stare seduti a fissare lo schermo del vostro Amiga), potete allora rimuovere la linea

```
#include "fastflood. h"
```

rimpiazzando l'unica chiamata a FastFlood( ) con la normale chiamata a Flood(). Se lo fate, potete probabilmente ridurre in maniera significativa lo stack, ma ricordate che le routine per i frattali sono fortemente ricorsive e richiedono di per se stesse una discreta quantità di stack.

Una nota sulla compilazione

Quando compilate AMOI, usate la libreria Fast Floating Point della Motorola, per favore! Le routine di generazione implicano molti complessi calcoli matematici e senza la libreria FFP potreste aspettare a lungo i vostri paesaggi. Se qualcuno volesse darmi un 68881, sarò molto felice di dirgli che tipo di miglioramento è in grado di generare.

Le istruzioni di AMOI per l'utente

La versione corrente (2.7) di AMOI è progettata per girare su un Amiga da 512 Kbyte e i risultati possono essere stampati (sebbene AMOI non abbia funzioni per stampare direttamente).

La generazione di un paesaggio si effettua in tre fasi distinte:

* creazione di una mappa delle curve di livello
* conversione della mappa in reticolo
* proiezione dei dati del reticolo in una immagine a due dimensioni

Le ultime due fasi sono interamente realizzate dal computer, richiedono solo che l'utente le faccia partire e non è richiesto nessun altro intervento. la prima fase (creazione di una mappa delle curve di livello) è un procedimento dell'utente, ed è in questo momento che la sua creatività deve dare i suoi frutti.

Lo schermo di AMOI

Quando si lancia AMOI, viene aperto uno schermo custom con dei gadget posti tutt'intorno. La parte principale dello schermo è occupata da un riquadro blu con bordo bianco, completamente vuoto: questa è l'area per disegnare in cui dovrete tracciare i contorni.
Sulla destra ci sarà una scelta di colori (quindici in tutto) che indicano le altitudini delle curve di livello, proprio come in una legenda di una normale carta geografica.
Infine, da qualche parte, vicino al fondo dello schermo, ci sarà un piccolo interruttore. Questo dovrebbe inizialmente mostrare la scritta Draw (disegno), che vi avverte che, se selezionate l'area per disegnare, il mouse comincerà a tracciare una sottile linea di contorno bianca.
L'interruttore Draw può essere selezionato per dare inizio al modo Fill (riempimento). Quando Fill è attivo, la selezione dell'area per disegnare causerà il riempimento di ogni figura posta sotto il puntatore del mouse, con il colore corrente. Si noti che quando si seleziona un colore per le curve di livello, il programma assume che la prossima cosa che intendete fare sarà il riempimento; così seleziona automaticamente il modo Fill.

Il disegno delle curve di livello

Il disegno di una mappa delle curve di livello implica il riempimento dell'intera area per disegnare con dei colori. Ogni colore rappresenta una differente altezza rispetto al livello del mare. I colori variano dal blu, che rappresenta il mare, al rosa, che rappresenta la più alta altitudine disponibile. La prima fase della creazione di una mappa

delle curve di livello consiste nel disegnare i confini delle diverse altitudini; queste altitudini corrispondono alle curve di livello in una normale carta topografica dell'Ordnance Survey. Per disegnare una curva di livello, si muova il mouse nel riquadro per disegnare. All'inizio, il tracciamento del mouse è disabilitato, così muovete il mouse all'inizio della vostra prima curva di livello e poi premete il pulsante sinistro del mouse. D'ora in poi, quando muoverete il mouse, esso lascerà una traccia dietro di sè. Quando avete completato una curva di livello, premete il pulsante sinistro del mouse per disabilitare il tracciamento e poi ripetete questo processo finché tutte le vostre curve di livello siano state disegnate.

Una volta che tutte le curve di livello siano state disegnate è necessario indicare l'altitudine delle varie regioni così definite. Per far ciò premete il pulsante sinistro del mouse sul gadget Draw/Fill. Ora, quando entrerete nell'area per disegnare e premerete il pulsante sinistro del mouse, la regione a cui state puntando verrà riempita con il colore relativo all'altitudine corrente. L'altitudine corrente è segnalata dalla posizione indicata dalla freccia bianca nell'area di selezione dei colori. L'altitudine può essere cambiata selezionando uno qualsiasi dei riquadri colorati.
Se usate accidentalmente il colore sbagliato, potete selezionare quello giusto e riempire nuovamente l'area. Potete passare dal disegno al riempimento un numero illimitato di volte, così potete costruire lentamente una mappa complessa, completando piccole aree una dopo l'altra. Quando avete finito di creare la vostra mappa delle curve di livello, conviene salvarla su disco per usi futuri.

I menu di AMOI

Lo schermo AMOI ha una piccola barra di menu divisa in due aree: Project e Generate. Il menu Project fornisce l'accesso a tutti gli I/O verso il disco (come il salvataggio delle già menzionate curve di livello) e alle onnipresenti opzioni About e Quit. Generate è molto più interessante, perché serve a dare inizio all'elaborazione delle informazioni delle curve di livello.
Ora che il sistema conosce i dettagli sulle curve di livello per il vostro paesaggio, deve "limare" i gradini fra un'altitudine e l'altra, impliciti in una mappa delle curve di livello. Per prima cosa, selezionate l'opzione Process del menu Generate. Se non avete ancora salvato la mappa delle curve di livello, vi sarà chiesto se intendete farlo ora. Allo schermo si sovrapporrà un riquadro contenente una lunga barra orizzontale. Questa è nota, solo per il motivo che suona bene, come "Decaying Alert" (allarme di decadimento) e indica il progredire delle operazioni.
La prima fase dell'opzione Process prevede la rimozione delle curve ed è particolarmente veloce. La seconda, la fase di interpolazione, dipende dal numero e dalla complessità delle curve di livello che avete disegnato; così piccole isole impiegheranno meno tempo per l'elaborazione, rispetto a valli e colline arrotondate e complesse.
La barra d'interpolazione è poi sostituita da un'altra barra a corrosione, dal titolo "Calculating" (che sta per "Calculating Aggregates", calcolo degli aggregati). Quando questa barra si è corrosa (e ciò avviene ad una discreta velocità su Amiga) la mappa è stata generata.
Potete ora salvare la mappa così generata su disco, per eliminare la necessità di rielaborare le curve di livello ogni volta che volete vedere il paesaggio. Lo si può fare con l'opzione Save Map del menu Project. I dati della mappa saranno salvati come un file .MAP. Una volta che il file è stato salvato potete generare una veduta del paesaggio.
Per vederlo, selezionate l'opzione in 32 colours che si trova sotto l'opzione View del menu Generate e dopo un certo tempo (e un Decaying Alert) apparirà la proiezione del paesaggio. Non potete vedere un paesaggio appena disegnato senza elaborarlo e così se rispondete "No" alla domanda "Shall we Process ?" ("Dobbiamo elaborare?"), ritornerete semplicemente al modo operativo normale.

Lo schermo del paesaggio

Quando genera un paesaggio, AMOI mantiene lo schermo su cui sta disegnando dietro lo schermo visibile, per ottimizzare le operazioni. Quando la generazione è conclusa, lo schermo è portato di fronte agli altri e si rende disponibile un nuovo menu. Il menu del paesaggio ha solo due opzioni, Title Bar (barra del titolo) e Exit (uscita), che sono entrambe autoesplicative. Per aiutarvi a gestire i vostri schermi, AMOI rivela automaticamente la barra del titolo dello schermo del paesaggio, quando abbandonate il modo di visualizzazione (questo vi assicura di non liberare altri schermi dietro di esso).

Altre opzioni

Le altre opzioni disponibili sulla barra di menu dello schermo principale sono molto semplici:

Clear (nel menu Project) pulisce l'area per disegnare, ristabilendo il colore blu del livello del mare. Vi viene chiesta una conferma.

Fractals (nel menu Generate) è un'opzione on/off che vi permette di disabilitare la fase di "frattalizzazione" nella generazione dei paesaggi. Con qualche sperimentazione, può dare un'idea abbastanza buona degli effetti degli algoritmi frattali, velocizzando notevolmente la produzione di paesaggi.

In Glorious HAM (dall'opzione View nel menu Generate) permette la generazione di paesaggi usando il modo grafico HAM (con molti colori in più). Questa opzione non è implementata nell'attuale versione di AMOI perché il codice di generazione HAM costituisce, di per se solo, un progetto estremamente significativo (che può formare la base di un articolo futuro). Vale la pena di notare che anche le routine HAM richiedono le funzioni di supporto contenute in fastpixel.o (con piccole modifiche).

(segue a pag. 50)

Il generatore di parser Bison/Yacc

Ancora un altro compilatore di compilatore... e altri animali!

Eric Giguère

La scienza della programmazione è progredita fino al punto in cui lo sviluppo dei programmi può essere aiutato e migliorato attraverso l'uso di strumenti e librerie specifiche. Lo sviluppo di compilatori non costituisce un'eccezione a ciò. Questo è il primo di due articoli che intendono costituire un'introduzione allo sviluppo di compilatori e un manuale elementare sul modo di usare uno degli strumenti per compilatori disponibili su Amiga... il generatore di parser Bison/Yacc.

Finiremo la nostra discussione nel prossimo articolo, usando Bison per costruire un programma non banale (e si spera anche utile) basato sulla tecnologia della compilazione. Si dovrebbe precisare a questo punto che questo articolo presuppone che voi non abbiate alcuna conoscenza di come lavora un compilatore, al di là di quanto potreste avere appreso usandone uno.

Traduttori, compilatori e interpreti

Un traduttore è un programma che prende una lista o uno script di istruzioni e le converte in una forma eseguibile dalla macchina. I traduttori sono normalmente classificati o come compilatori o come interpreti.

Un compilatore produce codice stand-alone; ciò significa che, una volta che le istruzioni siano state tradotte, il compilatore non è più necessario per la loro esecuzione.

Un interprete traduce ed esegue le istruzioni allo stesso tempo, senza produrre codice stand-alone.

Su Amiga la programmazione in C si fa con un compilatore (Manx o Lattice) e quella in Basic con un interprete (AmigaBASIC). La distinzione fra compilatori e interpreti ha tanto a che fare con il compromesso fra tempo di traduzione e velocità di esecuzione, quanto con la metodologia adottata. Un programma compilato sarà eseguito molto più velocemente della sua versione interpretata, perché si tratta di puro codice macchina, ma gli interpreti hanno un ciclo di sviluppo molto più veloce, perché il programma è tradotto in maniera incrementale invece che nel suo complesso.

I due tipi di traduttori hanno molto in comune, tuttavia, e così anche se il resto di questo articolo riguarderà i compilatori, molto di quanto viene detto si potrebbe applicare come tale agli interpreti. Un altro tipo di traduttore è l'assembler, che converte linguaggio assembly di basso livello nel suo puro equivalente in linguaggio macchina. Gli assembler sono in realtà proprio dei compilatori specializzati; interpreti assembler vengono usati solo per il debug o l'insegnamento.

Le parti di un compilatore

Concettualmente un compilatore può essere diviso in quattro parti: scanner, parser, tavola dei simboli e generatore di codice.
Lo scanner legge e trasforma il codice sorgente in token. I token sono i componenti più piccoli di un programma che hanno ancora un significato loro connesso. Token tipici sono le keyword, le stringhe, gli identificatori (nomi di variabile), operatori e numeri. Sebbene non siano particolarmente difficili da scrivere, è importante che gli scanner siano efficienti, perché essi effettuano molto I/O. Il lavoro dello scanner è anche chiamato analisi lessicale.

La tavola dei simboli contiene tutte le informazioni necessarie durante la compilazione del programma: nomi degli identificatori, livelli di visibilità, definizioni di tipo e così via. Può essere semplice come una lista concatenata o complessa come un insieme di tavole hash.
Il generatore di codice è la parte più ampia e complessa del programma, responsabile della traduzione del flusso di token forniti dallo scanner in linguaggio macchina. Il generatore di codice è normalmente composto a sua volta da più parti. Il generatore di codice del compilatore C della Manx, per esempio, genera linguaggio assembly e poi usa un assembler per generare codice macchina.

Uno schema comune è quello di avere un codice generato dal compilatore indipendente dalla macchina, che un secondo compilatore poi traduce in codice adatto ad una macchina specifica. Questa separazione dei compilatori in due parti favorisce la portabilità e permette un veloce sviluppo di altri compilatori.
Il parser controlla il compilatore. Esso analizza il flusso dei token che lo scanner gli invia e decide quale routine del generatore di codice vada chiamata. Il parser è anche conosciuto come analizzatore sintattico.

E' importante notare la differenza tra sintassi (l'ordine e la struttura dei token) e semantica (il loro significato). Un programma sintatticamente corretto non è necessariamente corretto semanticamente; il generatore di codice normalmente include un analizzatore semantico che determina se il codice che gli si è detto di generare significa qualcosa.
Per esempio l'espressione C ++ptr++ è corretta sintatticamente, ma le regole del C stabiliscono che solo uno degli operatori di incremento possa essere applicato ad una variabile. Il generatore di codice segnalerà questo come un errore.

I tipi di parser

I parser appartengono a due classi generali: table-driven e recursive-descent.

Un parser recursive-descent è quello in cui le regole sintattiche del linguaggio sono descritte usando procedure ricorsive. Essi "discendono" attraverso le regole finché non ne trovano una che gestisce il corrente flusso di token.
La tecnica recursive-descent è usata generalmente in parser "codificati a mano" oppure quando è richiesto un intenso lavoro di gestione degli errori (e possibilmente di correzione degli errori) e non verrà discusso in questa sede.

I parser table-driven usano un insieme di tavole generate, per controllare il compilatore. Le tavole dicono al parser driver quali sequenze di token sono valide e quali routine attivare quando una determinata sequenza è riconosciuta. Lo stesso parser driver può essere usato in molti differenti compilatori, solo le tavole devono essere cambiate.
Il metodi esatti usati da un parser table-driven non sono importanti nella nostra discussione. Questo tipo di parser è di solito generato dalla macchina mediante un programma chiamato un generatore di parser.

Yacc e Bison

Yacc è un generatore di parser sviluppato presso gli AT&T Bell Laboratories nei tardi anni '70. Yacc è un acronimo per Yet Another Compiler-Compiler (ancora un altro compilatore di compilatore), dal momento che questo è effettivamente ciò che è un generatore di parser.
Yacc è una parte standard del sistema di sviluppo Unix e ha raggiunto grande popolarità fra gli hacker Unix; i corsi sui compilatori alle università sono spesso svolti su macchine Unix usando Yacc e il C. Il solo problema è che Yacc può essere usato solo su macchine con licenza AT&T.

Nella prima parte degli anni '80, tuttavia Bob Corbett e altri hanno sviluppato un programma analogo a Yacc che hanno chiamato Bison (bisonte) (se non afferrate il motivo, cercate la parola "Yak" sul dizionario). Bison è un generatore di parser freeware che accetta lo stesso input di Yacc e genera tabelle simili. Bison è stato portato su molte macchine, una versione Amiga è disponibile sul Fish Disk #136 (si dovrebbe notare a questo punto che Bison non si trova nel public domain ed è stato inteso per lo sviluppo di software non commerciale o freeware).
D'ora in poi il termine Bison sarà usato per riferirsi sia al generatore di parser Bison che a quello Yacc.

Grammatiche del linguaggio

Per generare un parser per uno specifico linguaggio deve essere fornito a Bison un insieme di regole sintattiche: una grammatica. La grammatica descrive sia la sequenza dei token permessa che le azioni (routine) da eseguire quando sono riconosciute sequenze valide. Una grammatica Bison elenca, inoltre, altre informazioni che il generatore parser richiede, ma scenderemo nello specifico la prossima volta. Per ora consideriamo la grammatica Bison in un senso più generale.

Una grammatica è composta di simboli terminali, simboli non terminali e metacaratteri:

* Simbolo terminale è sinonimo di token
* Un simbolo non terminale rappresenta una sequenza di simboli terminali o non terminali. Ciascuna regola di una grammatica definisce la sequenza che costituisce un non terminale.
* I metacaratteri sono operatori come : e | che sono usati per definire le regole della grammatica.

Questi concetti si spiegano meglio con degli esempi. Consideriamo il linguaggio C. Una costante in C può essere un intero, un numero in floating point, un singolo carattere o l'identificatore di un'enumerazione. In una grammatica per il C codificheremmo:

```
constant: T_INTEGER;
constant: T_FLOAT;
constant: T_CHAR;
constant: T_ENUM;
```

Qui constant (costante) è non terminale, T_INTEGER, T_FLOAT, T_CHAR e T_ENUM sono terminali, mentre : e ; sono metacaratteri. Ogni regola grammaticale ha la forma:

```
LHS : RHS ;
```

dove LHS e un singolo non terminale e RHS è costituito da uno o più terminali o non terminali. In effetti una tale regola dice che "il non terminale etichettato LHS è un sinonimo per la sequenza chiamata RHS", una specie di espressione di assegnamento.
E' importante non confondere i metacaratteri con i token: per esempio, il punto e virgola qui sopra è usato per concludere una regola grammaticale e non per indicare che il punto e virgola fa parte dell'assegnamento.

Non c'è un'ovvia distinzione tra terminali e non terminali nella grammatica, se non per il fatto che ogni simbolo non terminale deve apparire almeno una volta come lato sinistro

di una regola grammaticale. E' utile, tuttavia, distinguerli visivamente fra di loro, usando stili differenti come è stato fatto qui sopra.
I nomi usati per terminali e non terminali devono essere unici e per chiarezza dovrebbero descrivere in qualche modo ciò che stanno definendo, come "costante", "espressione" e "istruzione".
Tornando al nostro esempio, vediamo che quattro regole diverse definiscono ciò che una costante può essere. Queste regole possono essere anche combinate fra loro, raggiungendo lo stesso effetto:

```
constant: T_INTEGER
        | T_FLOAT
        | T_CHAR
        | T_ENUM
        ;
```

Qui introduciamo un nuovo metacarattere | per le alternative. Così "constant" può essere una fra T_INTEGER, T_FLOAT, T_CHAR o T_ENUM. Il punto e virgola viene usato solo alla fine dell'ultima scelta.

Ora prendiamo in considerazione questo insieme addizionale di regole:

```
add_or_minus : constant
             | add_or_minus T_PLUS constant
             | add_or_minus T_MINUS constant
             ;
```

dove T_PLUS e T_MINUS sono i token per i segni più e meno rispettivamente. Mettiamo che lo scanner incontri l'input seguente:

```
3 + 5.0 - 6
```

Dopo l'analisi lessicale esso diventa:

```
T_INTEGER T_PLUS T_FLOAT T_MINUS T_INTEGER
```

Per ora non ci preoccupiamo del modo in cui il valore reale di ciascuno dei numeri viene immagazzinato, invece consideriamo solo i loro token. Quando lo scanner passa questi token al parser, questo comincerà ad applicare le regole descritte dalla grammatica, e pertanto ridurrà la sequenza di terminali quanto più è possibile.

Per prima cosa rimpiazza T_INTEGER e T_FLOAT con constant:

```
constant T_PLUS constant T_MINUS constant
```

Il parser non ha alcuna regola per rimpiazzare "constant T_PLUS" così rimpiazza "constant" usando la regola per "add_or_minus" (somma o sottrazione):

```
add_or_minus T_PLUS constant T_MINUS constant
```

Ora può ridurre "add_or_minus T_PLUS constant" usando una delle regole alternative per "add_or_minus":

```
add_or_minus T_MINUS constant
```

E di nuovo:

```
add_or_minus
```

La sequenza originale di cinque terminali è ora stata ridotta a un singolo non terminale. Il parser si ferma perché la grammatica non ha altre regole.
L'obiettivo del parser è quello di usare le regole della grammatica per ridurre il flusso dei token a un singolo non terminale, il simbolo obiettivo. Quando raggiunge questo simbolo il parser ha concluso e il compilatore si ferma.

E' importante essere sicuri che nessuna delle vostre regole grammaticali generi dei conflitti: se due regole differenti usano la stessa sequenza di simboli sul lato destro, il parser non saprà quale scegliere. Una grammatica di questo tipo è chiamata una grammatica ambigua.
Sebbene Bison può essere impostato per scegliere tra regole conflittuali, è sempre meglio evitare le ambiguità.

Azioni ed errori sintattici

Se tutto ciò che il parser fa è rimpiazzare i terminali con i non terminali, non sarebbe molto utile, se non come programma di controllo sintattico. Ogni regola, tuttavia, può avere associato un insieme di azioni; queste usano l'informazione fornita dal parser per generare codice. Le azioni vengono effettuate quando è necessaria una riduzione (una sostituzione) e saranno discusse in dettaglio in un prossimo articolo.
Un errore sintattico avviene quando il parser non può trovare una regola per ridurre il flusso di input corrente.

Il prossimo articolo

E' tutto per la teoria. La prossima volta studieremo a fondo i particolari sull'uso di Bison per generare parser e come usare quei parser nei vostri programmi. Creeremo una nostra grammatica, uno scanner e delle routine per le azioni; dovrebbe essere interessante!

Nel frattempo, date un'occhiata all'Appendice A del libro di Kernighan e Ritchie, The C Programming Language, per vedere come appare nei dettagli la grammatica C.

Linguaggio Assembly

Parte 4 - Uso e gestione della memoria

Jim Butterfield

Jim Butterfield non ha bisogno di presentazioni. Il suo nome costituisce un punto di riferimento per gli utenti Commodore di tutto il mondo. Il suo interesse per i computer risale al KIM-1 da 1K. L'enciclopedica competenza di Jim sui prodotti Commodore è testimoniata dai suoi articoli, libri, lezioni e persino programmi televisivi.

L'accesso alla memoria

Abbiamo fatto un bel po' di lavoro con dati e file. Finora, però, siamo stati piuttosto avari a proposito di memoria. L'effetto collaterale è stato un uso elaborato dei registri; abbiamo dovuto tracciare un piano molto preciso per l'uso dei registri al fine di mantenere in essi la massima quantità possibile di informazioni.

Ciò dà velocità, naturalmente. Ma c'è un limite allo sviluppo di questa tecnica. E' tempo di indagare un po' la memoria e di mostrare l'uso più tipico delle variabili. Non potete servirvi da soli della memoria, in un computer multitasking come Amiga; altri processi potrebbero, nel frattempo, usare quella specifica parte della memoria. Dobbiamo "avanzare una richiesta formale" per la memoria di cui abbiamo bisogno.

Osserveremo più da vicino lo stack, puntato dal registro A7; si tratta di un luogo da cui è facile ricavare quantità ragionevolmente piccole di memoria di lavoro. E richiederemo aree più grandi di memoria a Exec, mediante la funzione AllocMem( ).
Sia che prendiamo la memoria dallo stack, che dall'area generale (spesso chiamata heap, mucchio, ammasso), dobbiamo fare attenzione a restituire ogni pezzo di memoria che abbiamo preso in prestito. Se non lo facessimo, la memoria potrebbe essere perduta per sempre, almeno fino al prossimo reset di sistema; e nel caso peggiore, una gestione cattiva della memoria potrebbe provocare una visita del Guru.

Il programma

Il programma rfile (da read file, leggere file) leggerà un file e lo invierà sullo schermo. Immane compito..., l'ultima volta abbiamo fatto qualcosa di più complicato; e inoltre TYPE svolge lo stesso lavoro. Ma noi affronteremo il compito in un modo leggermente differente; e il programma non solo fornirà lo scenario per il futuro e più sofisticato lavoro sui file, ma anche ci darà un piccolo vantaggio pratico.

Il nostro obiettivo è leggere l'intero file in un colpo solo, ammesso che ci sia memoria sufficiente. Questo sarà molto più veloce che dieci-caratteri-alla-volta, il metodo usato nel nostro esempio precedente. Esso, inoltre, dischiude un orizzonte di nuove possibilità. Non vogliamo andare più in là questa volta, ma si consideri che se un intero file viene caricato in memoria, si è già in una buona posizione per compiti quali l'ordinamento o la ricerca.

Di nuovo, il nostro programma potrà essere lanciato solo da CLI. Il nome del comando sarà seguito dal nome del file, per esempio:

```
rfile s:startupII
```

Tratteremo, un giorno, programmi lanciabili da Workbench, ma ciò implica un livello più complesso di programmazione; per il momento, continueremo a perfezionare la nostra destrezza nella programmazione e la nostra familiarità con il sistema.

Il funzionamento

Per leggere l'intero file, dobbiamo prima ottenere abbastanza memoria per contenere tutti i dati. Dobbiamo effettuare una chiamata alla subroutine AllocMem( ) (alloca la memoria) e più tardi una chiamata a FreeMem( ) (libera la memoria) per restituire lo spazio di memoria. Ma come possiamo sapere quanta memoria richiedere? Il file potrebbe avere qualsiasi lunghezza; dobbiamo chiedere informazioni sulla sua lunghezza.

Il modo migliore per stabilire la lunghezza di un file è richiedere il suo FileInfoBlock (blocco di informazioni sul file), che contiene molte voci che riguardano il file. La figura 1 mostra come risieda in memoria un FileInfoBlock; vedete che 124 byte più in basso, c'è una voce chiamata fib_Size che esprime la lunghezza in byte del file, proprio ciò che cerchiamo.
Prima di ottenere il FileInfoBlock tuttavia, dobbiamo riservargli un posto in memoria. Ciò significa un'altra chiamata

ad AllocMem( ) per i 260 byte, e più tardi il loro rilascio mediante FreeMem( ). E' una procedura semplice e chiara.

Otteniamo il FileInfoBlock chiamando la funzione Examine( ) della libreria di subroutine del DOS. Questa routine si aspetta che gli si fornisca l'indirizzo di memoria in cui può porre il FileInfoBlock. Perfetto: sappiamo come ottenere quel pezzo di memoria. Ma essa vuole anche, per il file, una cosa chiamata lock (serratura, lucchetto). Come ottenerla?

Ricaviamo il lock per il file chiamando la routine DOS Lock(). Chiamiamo questo un lock in lettura (potreste pensare ad esso come ad un dispositivo di sincronizzazione, "interlock", in inglese).
Questo significa che altri programmi possono leggere il file nello stesso momento in cui noi lo leggiamo, ma nessuno potrà scrivere nel file finché non abbiamo terminato di leggerlo.
Ciò completa l'elenco di cose da fare. Rivediamo l'intera procedura, mettendo le parti nel giusto ordine.

Lock: ricavare un lock in lettura sul file.
AllocMem: ricavare un blocco di memoria per contenere il FileInfoBlock.
Examine: caricarvi il FileInfoBlock per il file.
AllocMem: ricavare sufficiente memoria per contenere tutto il file.
Open, Read e Close: leggere l'intero file.
FreeMem: restituire il blocco di memoria per l'intero file.
FreeMem: restituire il blocco di memoria per il FileInfoBlock.
UnLock: rilasciare il file.

E' una bella sequenza di eventi, ma non è difficile da seguire. Potete notare come gli eventi si annidino, così che la linea iniziale e quella finale si corrispondono reciprocamente, e così via.

Un po' di background

Useremo parte dello stack per le nostre variabili di lavoro. La quantità di memoria di cui abbiamo bisogno sarà riservata mediante il comando LINK. Alla fine del programma, avendo terminato con quest'area di memoria, dobbiamo stare attenti a restituirla con il comando UNLK. La figura 2 mostra lo stack di cui ci impadroniamo, usando il comando LINK e come usiamo quest'area. In questo esercizio useremo i valori di offset, come sono visibili nella figura, per indirizzare le parti della memoria che vogliamo usare.

In futuro, usando il comando EQU (EQUate, equivale) per dare nomi simbolici a questi offset, otterremo un codice sorgente più autoesplicativo.
Si noti che qui è implicata un'operazione parallela. I comandi LINK e UNLK allocano memoria sullo stack, mentre le routine AllocMem( ) e FreeMem( ) fanno lo stesso tipo di operazione, tranne per il fatto che l'allocazione riguarda la memoria generale. In entrambi i casi, dobbiamo fare attenzione a restituire la memoria che abbiamo preso a prestito.

Entrambi i sistemi allocano memoria ad uso del programmatore, ma ci sono delle differenze importanti. Lo stack può avere solo alcune migliaia di byte, pertanto non possiamo impadronirci, mediante esso, di una smisurata quantità di memoria. LINK e UNLK sono comandi facili e veloci, così per piccole quantità di memoria sono proprio ciò che ci serve.

AllocMem( ) richiede uno maggiore sforzo di preparazione, ma offre molti vantaggi. Possiamo chiedere molta memoria, se lo desideriamo. Possiamo richiedere tipi speciali di memoria (memoria CHIP o FAST, per esempio). E una cosa ancora: ci sarà fornita memoria allineata alle longword. Parleremo di questo fra un momento; questo allineamento è utile per ricevere un FileInfoBlock.

Figura 1

➡	Vecchio valore di A4	⬅ Nuovo valore di A4
-$4(A4)	Puntatore alla linea di comando	
-$8(A4)	Lunghezza della linea di comando	
-$C(A4)	Puntatore alla base della libreria DOS	
-$10(A4)	File handle di output	
-$14(A4)	Lock in lettura del file	
-$18(A4)	Puntatore al FileInfoBlock	
-$1C(A4)	Lunghezza del file	
-$20(A4)	Puntatore all'area dati del file	
SP dopo LINK ➡		

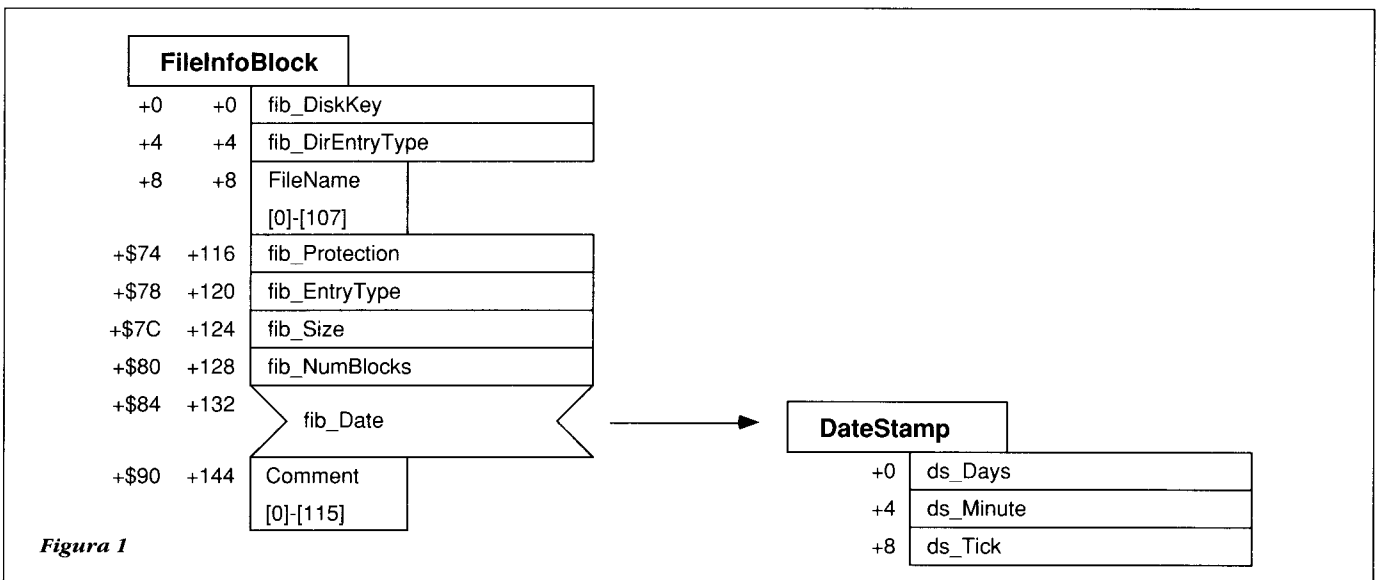
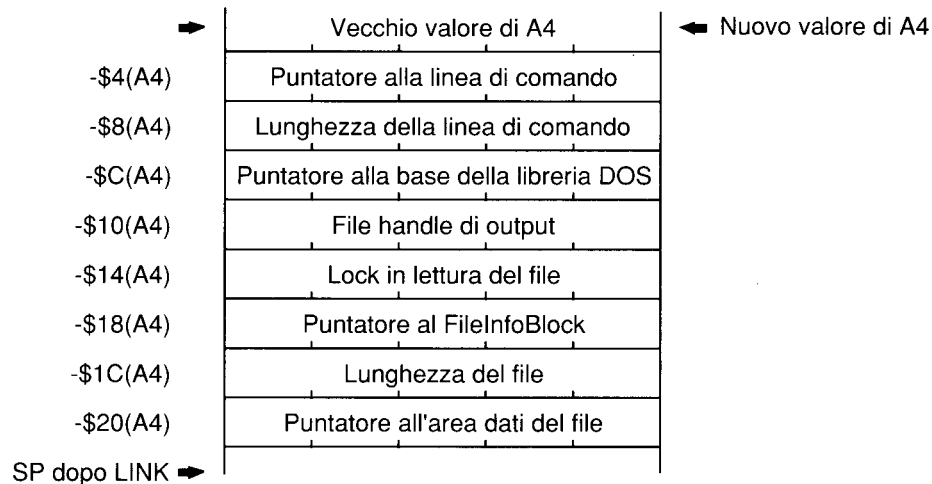



Figura 2

La struttura FileInfoBlock

Una struttura è una parte di memoria organizzata in un certo modo particolare. La struttura FileInfoBlock è un area di memoria organizzata per contenere informazioni riguardanti un file o una directory. Abbiamo già menzionato questa struttura: la si vede nella figura 2. Potete ricavare i dettagli di una struttura dai file include, se li avete.

Il FileInfoBlock si troverà nel file dos.h [per il linguaggio C, dos.i per l'Assembly, N.d.T.]. Per il momento, useremo gli offset numerici.
Non sappiamo dove starà questa struttura finché non allochiamo memoria per contenerla. Questo è facile da realizzare nel programma. Un registro terrà l'indirizzo della memoria allocata e useremo questo indirizzo nel modo indiretto per accedere al blocco.
Più precisamente, possiamo raggiungere ogni parte della struttura che desideriamo mediante un "offset"; il nostro modo di indirizzamento sarà indiretto con offset.

Una prima occhiata ai BPTR (puntatori BCPL)

Il FileInfoBlock lo va a prendere per noi, in quella parte del sistema operativo di Amiga chiamato DOS (Disk Operating System). Il DOS fu scritto in un linguaggio chiamato BCPL. Non avete bisogno di conoscere questo linguaggio, ma esso ha un interessante effetto collaterale: preferisce gli indirizzi divisibili esattamente per quattro.

Il DOS spesso lavora con indirizzi che sono stati divisi per quattro. Tali indirizzi, contenuti in longword, sono detti BPTR (puntatori BCPL).
Incontrerete altri BPTR, man mano che vi inoltrerete in profondità nei misteri del DOS, come nelle sequenze di startup dal Workbench. In quel momento, potreste ottenere un valore BPTR; dovrete moltiplicarlo per quattro, per ricavare l'indirizzo reale. Non è difficile, e alcune persone pensano che l'allineamento alle longword sarà un vantaggio quando ci si muoverà verso il microprocessore 68030. Noi abbiamo già usato dei BPTR senza conoscere la loro identità: un file handle è un BPTR.

A causa del fatto che il DOS spesso usa i valori allineati alla longword, dobbiamo avere una particolare attenzione quando usiamo Examine( ). Per conservare il FileInfoBlock, dobbiamo fornire un indirizzo che deve essere allineato alla longword. In altre parole, il FileInfoBlock deve iniziare a un indirizzo che si divide esattamente per quattro.

Nessun problema. AllocMem( ) vi dà sempre un indirizzo di questo tipo (allineato alla longword). E' questo il motivo per cui la usiamo per il relativamente piccolo FileInfoBlock. Sembrerebbe facile impadronirci di 260 byte dello stack; ma se l'avessimo fatto, non avremmo potuto essere sicuri che il primo indirizzo si sarebbe diviso esattamente per quattro.

Il codice

Usate il vostro editor preferito per creare un nuovo file chiamato rfile.asm. Continueremo le spiegazioni attraversando il codice.
Dapprima le cose principali. Un titolo seguito da alcune lineee XREF: alcune di queste sono già state usate in precedenza, e alcune sono nuove.

```
;
; Carica e stampa un file - Jim Butterfield.
; Chiamate alla libreria Exec
  xref _LVOAllocMem ;-$C6
  xref _LVOFreeMem  ;-$D2
  xref _LVOCloseLibrary ;-$19E
  xref _LVOOpenLibrary  ;-$228
; Chiamate alla libreria DOS
```

```
xref _LVOOpen       ;-$1E
xref _LVOClose      ;-$24
xref _LVORead       ;-$2A
xref _LVOWrite      ;-$30
xref _LVOOutput     ;-$3C
xref _LVOLock       ;-$54
xref _LVOUnLock     ;-$5A
xref _LVOExamine    ;-$66
```

I valori che seguono sono adoperati da specifiche routine. Li rendiamo simboli, nella speranza di rendere le routine più facili da interpretare.
ACCESS_READ sarà usato da Lock( ); MODE_OLDFILE è richiesta da Open( ). INFOSIZE è la lunghezza di un FileInfoBlock e SIZEPOINTER è il punto, entro il blocco, in cui troveremo la lunghezza del file che stiamo investigando.

```
ACCESS_READ        equ -2
MODE_OLDFILE       equ 1005
INFOSIZE           equ 260
SIZEPOINTER        equ 124
```

Ora ci accingiamo a ricavare dello spazio sullo stack e a porre dei valori di lavoro nella memoria. Abbiamo bisogno di 8 longword (32 byte) per i valori che abbiamo previsto. Otterremo questi 32 (20 esadecimale) byte usando LINK; l'uso di questo "quadro" è raffigurato nella figura 2. Non ci si preoccupi del fatto che i valori numerici sulla tavola siano negativi. Lo spiegheremo tra un attimo.

Nessuno degli elementi ha ancora un valore. Li stabiliremo più avanti. Di fatto, questi elementi non hanno ancora una locazione in memoria. Chiediamo dunque allo stack di fornirci 32 byte per contenere questi numeri.
A questi elementi, normalmente, si attribuiscono nomi simbolici, così che si possa usare (con le EQU opportune) l'espressione OutHandle(A4) invece di -$10(A4). Il primo modo sembra indubbiamente migliore. Questa volta, tuttavia, voglio fornire i numeri grezzi, per aiutarvi a vedere come funziona il meccanismo.

```
;-- Preparazione iniziale:
Startup
  link    a4,#-$20
```

Lo stack ci ha dato 32 (20 esadecimale) byte; il registro A4 ora punta proprio sopra la cima di quest'area. La longword più alta (una longword è lunga 4 byte, ovviamente) partirà 4 byte al di sotto dell'indirizzo di A4 e giungerà fino al byte immediatamente al di sotto di A4.
Se noi poniamo delle parentesi attorno ad A4, creiamo un indirizzo indiretto: l'indirizzo al quale A4 punta. Se noi ci riferiamo a un valore che lo precede, per esempio -$4(A4), il valore funziona come uno spiazzamento che aggiusta l'indirizzo indiretto verso l'alto o verso il basso. Per esempio, se A4 contiene l'indirizzo $2468, l'espressione -$4(A4) si riferisce al contenuto dell'indirizzo $2464.
C'è abbastanza spazio sullo stack? Nel caso di 32 byte, il pericolo è poco. Per blocchi più grandi di stack, potremmo aver bisogno di fare test più sofisticati... ma non ora.

```
  move.l  a0,-$4(a4);Indirizzo della linea di comando
  move.l  d0,-$8(a4); Lunghezza della linea di comando
```

Viene salvato il contenuto di A0 e D0. Svolgiamo il solito compito di aprire la libreria DOS.

```
  lea     dosname(pc),a1     ; Nome 'dos.library'.
  clr.l   d0                 ; Qualsiasi versione (0).
  move.l  $4,a6              ; Usiamo la libreria Exec.
  jsr     _LVOOpenLibrary(a6);Apriamo la libreria DOS
  move.l  d0,-$C(a4)         ; Ricordiamo il ptr a DOSBase.
  beq.s   StartupQuit        ; il DOS non si apre!
```

Si noti che l'istruzione MOVE.L imposta il flag di zero che viene controllato dal seguente BEQ. Se giungiamo a questo punto senza problemi, possiamo chiamare DOSinit.

```
  bsr.s   DOSinit
```

Quando DOSinit ritorna, dobbiamo effettuare delle operazioni di chiusura: chiudere il DOS e restituire (mediante UNLK) la memoria presa in prestito dallo stack.

```
  move.l  -$C(a4),a1     ; DOSBase in A1,
  move.l  $4,a6          ; usiamo poi la libreria Exec
  jsr     _LVOCloseLibrary(a6) ; per chiudere la
                                 libreria DOS.
StartupQuit
  unlk    a4
  rts                          ; Fine del programmma
```

Se la prima fase ha successo, raggiungiamo DOSinit. Facciamo il solito lavoro di ricavare l'Output handle, riponendolo nella nostra tavola sullo stack. Come al solito, poniamo uno zero in coda al nome del nostro file ed eliminiamo gli spazi all'inizio.

```
  ;-- Ricaviamo l'handle di output del CLI:
DOSinit
  move.l  -$C(a4),a6     ; impostiamo la libreria DOS
  jsr     _LVOOutput(a6) ;chiediamo l'handle per
                          l'output,
  move.l  d0,-$10(a4)    ; e poi ricordiamolo.
  ;
  ;--Modifichiamo il nome del file in una stringa di
  tipo C:
checklen
  move.l  -$4(a4),a3 ;indirizzo della linea di comando
  move.l  -$8(a4),d3; lunghezza della linea di comando
  move.b  #0,-$1(a3,d3.W) ;zero binario alla fine
                        skipspc
  cmp.b   #$20,(a3)+
  beq.s   skipspc
  subq.l  #1,a3
  move.l  a3,-$4(a4)   ; salviamo l'indirizzo corretto
```

Ora le nuove operazioni. Il manuale del DOS indica che il formato del comando Lock() deve essere:

```
lock = Lock(nome, mododiaccesso)
D0          D1    D2
```

Per un lock in lettura, il nostro modo di accesso avrà il valore di -2. In memoria, apparirà come il valore esadecimale FFFFFFFE.

```
;ricaviamo un lock sul file, se possibile:
move.l  a3,d1
moveq   #ACCESS_READ,d2;chiediamo un lock in lettura
jsr     _LVOLock(a6)
move.l  d0,-$14(a4);riponiamo il puntatore al lock
beq     DosQuit  ; non possiamo disporre del lock
```

Ci sono ragioni piuttosto rare (come quella che qualcuno sta scrivendo sul file) per le quali potremmo non riuscire ad ottenere il lock sul file. C'e una ragione molto probabile: avete sbagliato nello scrivere il nome del file o avete indicato il path errato. Per risparmiare spazio e digitazione, non inviamo un messaggio di errore se accade una di queste cose; più semplicemente, terminiamo il programma.

Se abbiamo ottenuto il lock, siamo pronti a ricavare il blocco di informazioni sul file. Chiediamo 260 byte di memoria. Dobbiamo cambiare la libreria, dal momento che il AllocMem( ) è una funzione Exec.
Il ROM Kernel Manual indica nel seguente modo il formato per AllocMem( ):

```
bloccodimemoria = AllocMem(lunghezza, tipo)
 D0                        D0         D1
```

```
;Chiediamo la memoria per contenere un FileInfoBlock:
  move.l  $4,a6          ; la chiamata è per Exec
  moveq   #0,d1          ; qualsiasi tipo di memoria
  move.l  #INFOSIZE,d0   ; lunghezza del blocco che ci
                           occorre
  jsr     _LVOAllocMem(a6) ; andiamo a richiederla
  move.l  d0,-$18(a4);salviamo l'indirizzo del blocco
  beq.s   DropLock       ; terminiamo se non c'è memoria
```

Il vostro Amiga dovrebbe essere veramente stipato se non riuscisse a trovare 260 miseri byte di memoria; ma, in ogni caso, il codice per controllarlo è presente. Il tipo di memoria è zero; in altre parole, qualsiasi tipo di memoria: CHIP o FAST. Ora torniamo alla libreria DOS per Examine( ).
Il Manuale dell'AmigaDOS mostra così il formato di Examine( ):

```
successo = Examine(lock, FileInfoBlock)
  D0               D1    D2
```

```
; Ora ricaviamo le informazioni sul file:
  move.l  -$C(a4),a6    ; impostiamo la libreria DOS
  move.l  -$14(a4),d1   ; lock
  move.l  d0,d2         ; indirizzo del FileInfoBlock
  jsr     _LVOExamine(a6) ; andiamo a ricevere i dati
; Assumiamo che Examine non fallisca mai.  Ricaviamo
la lunghezza:
  move.l  d2,a2         ; indirizzo del FileInfoBlock
  move.l  SIZEPOINTER(a2),-$1C(a4);lunghezza del file
  beq.s   FreeInfo
```

Una volta che abbiate un buon lock è difficile immaginare che Examine( ) fallisca, sebbene la documentazione mostri che tale risposta è possibile. Potreste avere un problema differente, invece il file potrebbe avere una lunghezza pari a zero. Potrebbe essere un file vuoto, ma è molto più probabile il fatto che avevamo un lock su una directory piuttosto che su un file.

Abbiamo stabilito la lunghezza del file. Siamo pronti a tornare ad Exec per eseguire un'altra AllocMem( ). Abbiamo bisogno di tenere una contabilità attenta, dal momento che ci sono blocchi da restituire e un lock su un file da rilasciare.

```
; OK,sappiamo la lunghezza.Chiediamo un blocco di
quella grandezza
  move.l  $4,a6           ; questo per chiamare Exec
  moveq   #0,d1           ; qualsiasi tipo di memoria
  move.l  -$1C(a4),d0 ;lunghezza del blocco di memoria
  jsr     _LVOAllocMem(a6)  ; richiediamolo
  move.l  d0,-$20(a4);salviamo l'indirizzo del blocco
  beq.s   FreeInfo       ; terminiamo se non c'è memoria
```

E' possibile trovare un file molto grande, troppo grande per la memoria che abbiamo a disposizione. In questo caso, non possiamo svolgere il lavoro e il programma termina. Ora, torniamo al DOS e leggiamo l'intero file.

```
; Evviva! Ora siamo pronti a leggere il file:
  move.l  -$C(a4),a6     ; impostiamo la libreria DOS
  move.l  -$4(a4),d1     ; puntatore al nome del file
  move.l  #1005,d2     ; MODE_OLDFILE (per la lettura)
  jsr     _LVOOpen(A6)
  move.l  d0,d6            ; file handle di input
  beq.s   FreeFileMem      ; non è valido, si termina
; Leggiamo l'intero file in un colpo solo:
  move.l  d6,d1            ; file handle di input
  move.l  -$20(a4),d2 ; indirizzo del buffer di input
  move.l  -$1C(a4),d3 ; lunghezza del buffer di input
  jsr     _LVORead(A6)     ; leggiamolo
  move.l  d6,d1            ; usiamo l'handle...
  jsr     _LVOClose(A6)    ; ...per chiudere il file
```

Open( ), Read( ) e Close( ). Il file ha svolto il suo compito e tutti i dati stanno ora in memoria.

```
  bsr.s   MainJob          ; il compito principale
```

Il compito principale è stato svolto. Ora restituiamo i due blocchi di memoria e liberiamo il lock sul file. Spero che notiate come la tabella di valori sullo stack, indicizzata

mediante A4, renda più semplice il lavoro di contabilità. Il ROM Kernel Manual indica così il formato di FreeMem():

```
FreeMem(bloccodimemoria, lunghezza)
        A1               D0

FreeFileMem
  move.l  $4,a6       ; impostiamo la libreria Exec
  move.l  -$20(a4),a1; indirizzo del blocco di memoria
  move.l  -$1C(a4),d0   ; lunghezza del blocco
  jsr     _LVOFreeMem(a6)      ; rilasciamo il blocco
FreeInfo
  move.l  $4,a6       ; impostiamo la libreria Exec
  move.l  -$18(a4),a1; indirizzo del blocco di memoria
  move.l  #INFOSIZE,d0  ; lunghezza del blocco
  jsr     _LVOFreeMem(a6)      ; rilasciamo il blocco
```

Il manuale DOS mostra che il formato del comando UnLock() deve essere:

```
UnLock(lock)
       D1

DropLock
  move.l  -$C(a4),a6 ; impostiamo la libreria DOS
  move.l  -$14(a4),d1   ; rilasciamo il lock
  jsr     _LVOUnLock(a6)
DosQuit  rts              ; usciamo dal programma.
```

Il nostro compito principale è qualcosa di deludente. A questo punto potremmo ordinare, indicizzare o fare altre cose meravigliose. Invece, mandiamo semplicemente in output l'intero file. Un solo comando Write() invia l'intero file. Spero che abbiate scelto un file che contenga del testo, perché un Control-C non interromperebbe il programma in questo caso.

```
; Il file è nel buffer. Scriviamolo in un colpo solo!
; Il  registro A6 contiene DOSBase
MainJob
  move.l  -$10(a4),d1   ;file handle
  move.l  -$20(a4),d2   ;buffer di output
  move.l  -$1C(a4),d3   ;lunghezza
  jsr     _LVOWrite(A6)
  rts

dosname  dc.b           'dos.library',0
  end
```

Il prodotto finale

Assemblate ed eseguite il link del programma: vi accorgerete che rfile stamperà velocemente un file sullo schermo. Niente di nuovo, sembra, ma avete avuto la possibilità di esaminare alcuni aspetti nuovi della gestione dei file e della programmazione.
C'è un trucco ulteriore, tuttavia: la copia dei file. Anche se un file non è stampabile, possiamo mandare delle copie, ovunque ci piaccia, usando la ridirezione. Volete copiare un file chiamato CANE in DF1:GATTO ? Il comando COPY svolge il lavoro, ma voi potreste anche tentare:

```
rfile >DF1:GATTO CANE
```

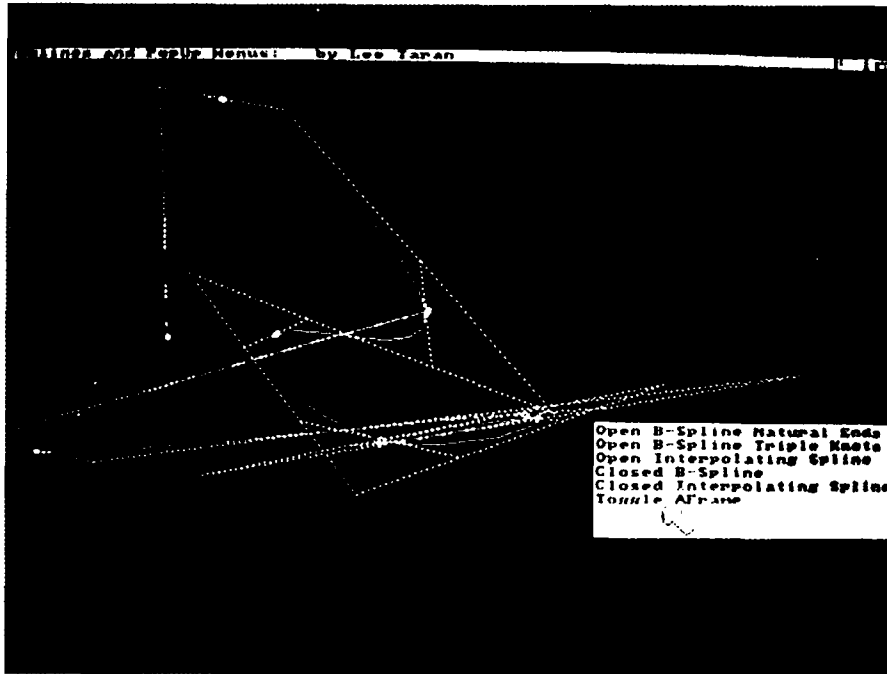
Il compito sarà svolto più rapidamente, rispetto al comando COPY, se avete RAM sufficiente a contenere l'intero file. Ed è interessante sapere che potete effettuare una copia con un programma che occupa poche centinaia di byte.

(segue da pag. 41)

AMOI

Conclusioni

La versione di AMOI per Amiga è stata scritta per il Lattice C versione 5.0 e il sorgente occupa attualmente più di 55 Kbyte, il che non è poco. Esso si trova, assieme all'eseguibile, sul dischetto che accompagna la rivista.
Ciò che speriamo è che alcune delle tecniche usate nel programma (come gestione di file IFF, controllo dei menu, proiezione tridimensionale, rimozione dei piani posteriori, ombreggiatura delle superfici, "frattalizzazione" e interpolazione) vi siano utili. E' sorprendente quanto spesso gli algoritmi di ombreggiatura delle superfici possano essere usate, per esempio per trasformare un cupo grafico tridimensionale in una vivida immagine ombreggiata dinamicamente, all'altezza dei migliori programmi televisivi.

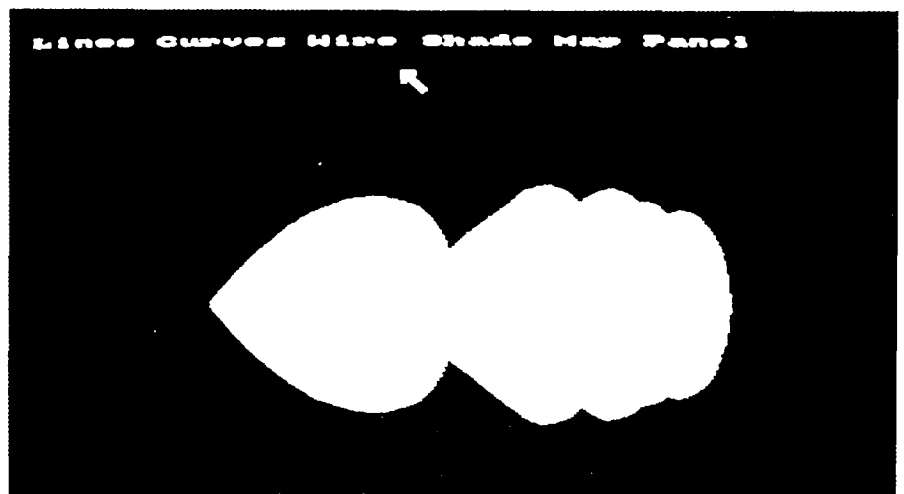
Se sarete affascinati dalla creazione di paesaggi personalizzati, come lo siamo noi, potrebbe interessarvi sapere che la versione 3 sarà presto disponibile presso di noi; oltre ad avere un'interfaccia utente più gradevole, grazie a Deluxe Paint, essa implementa sia i paesaggi in HAM, che una sorgente di luce definibile dall'utente.
Transactor ci ha gentilmente permesso di usare questo articolo per pubblicizzare la versione 3.0 che costa 4.99 sterline senza disco. A questo prezzo otterrete il codice sorgente C e Assembler, il codice oggetto (compilato con le opzioni di debug abilitate, per permettere analisi più stringenti), il codice binario e molti paesaggi esemplificativi.
Mi dispiace, ma non abbiamo la possibilità di accumulare riserve di dischi, così al fine di mantenere tempi di risposta veloci, abbiamo deciso di stabilire un prezzo più basso e di chiedere a voi di fornire il disco (che sarà esattamente quello che tornerà indietro!). Sarebbe anche un grande aiuto se riusciste a formattarlo (OFS o FFS). Si prega di fare assegni intestati a Danny Ross e di inviarli a: 54 Fore Street, Bradnich, EXETER, Devon EX5 4NN -England

programmi o dei comandi DOS.
La versione originale fa parte del set di comandi di Unix, il noto sistema operativo, e questa versione per Amiga la riproduce abbastanza fedelmente.
Per indicare ad Amicron gli orari in cui richiamare determinati programmi, è necessario scrivere una tabella, all'interno della quale viene indicata, per ogni programma che si desidera utilizzare, la data e l'ora.
E', inoltre, possibile indicare un certo range di date e orari, in modo da richiamare un determinato programma anche più volte al giorno e in diversi giorni dell'anno.
La tabella contenente tutti i dati relativi ai vari programmi, dovrà poi essere conservata su disco o nella RamDisk, in modo da essere accessibile ad Amicron in qualsiasi momento, senza bisogno di sostituire dischetti. Le applicazioni di AmiCron sono tra le più svariate e utili, dall'orologio che appare solo ogni cinque minuti, al backup automatico e periodico di dati contenuti in un file importante.
Per maggiori informazioni riguardo il formato della tabella dei task e l'utilizzo preciso di AmiCron riferitevi al file documento presente nella directory utility.

• Grafica

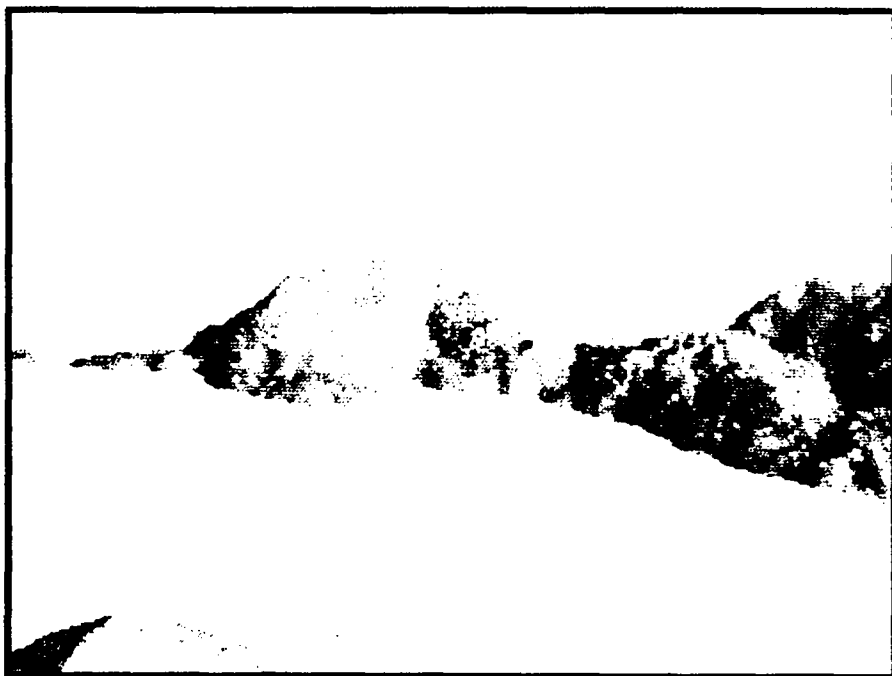
Spline

Questo suggestivo programma di grafica, serve a tracciare delle linee curve determinate da un certo numero di punti posti su di un pianò; in termine tecnico queste curve vengono chiamate "SPLINE" e, pur avendo applicazioni tecniche molto importanti, possono risultare anche molto belle e decorative, ed è questo l'aspetto che risalta maggiormente dal programma.
Una serie di opzioni appaiono quando si clicca con il tasto sinistro su uno dei punti dello spline che appare inizialmente sullo schermo, permettendovi di aggiungere o togliere dei punti.
Per modificare la natura dello spline è, invece, sufficiente cliccare il tasto sinistro del mouse in qualsiasi punto dello schermo per far apparire una nuova serie di opzioni.
Una spiegazione più dettagliata delle caratteristiche più particolari del programma è contenuta e nel file documento presente nella directory "grafica".

Surf

Anche Surf, come Spline, è un'applicazione delle curve Spline alla grafica computerizzata; in questo programma, però, una volta determinata una ben precisa spline, è possibile farla ruotare di 360 gradi su se stessa per creare un'immagine tridimensionale. Il risultato è, quindi, una superficie generata da una rivoluzione completa della spline rispetto ad una retta.
Il rendering di tale superficie può essere effettuato in vari colori e in varie risoluzioni, e presenta uno speciale effetto di ombreggiatura

che rende l'immagine molto suggestiva e realistica.
E' addirittura possibile, con l'ausilio di un buon programma di grafica HAM, fondere le tre schermate ottenibili utilizzando una delle tante opzioni di Surf, in un unica schermata a 4096 colori, che sicuramente rappresenterà con impressionante realtà l'oggetto ottenuto.
A questo punto i risultati dipendono solo dalla vostra fantasia, e con un po' d'ingegno potrete ottenere immagini molto decorative e piacevoli.
Per una descrizione dettagliata delle tante opzioni di cui il programma è dotato si rimanda il lettore al file documento presente nella directory "grafica".

Scenery

Questo programma di grafica sfrutta i frattali per creare dei paesaggi con monti e laghi veramente molto realistici; appena caricato, il programma comincia subito la tracciatura di un paesaggio creato, seguendo dei parametri di default.
Per completare il disegno Amiga puo impiegare anche una decina di minuti ma il risultato è sorprendente, tanto che sembra di trovarsi di fronte ad una immagine digitalizzata di un vero paesaggio di montagna; ma scenery ha ancora in serbo delle sorprese: è possibile modificare tutti i parametri che determinano l'altezza delle montagne, la grandezza del "lago" presente nel paesaggio, la direzione dalla quale arriva la luce per generare l'effetto di ombreggiatura e ray-tracing, e altri parametri di minore importanza che potrete scoprire voi stessi.
Come è immaginabile è possibile modificare il "seme" dei numeri random necessari a generare il paesaggio vero e proprio, cioè a determinare la posizione dei monti e la loro forma.
Con un po' di pazienza e di tempo è possibile ottenere da Scenery risultati stupefacenti e sempre diversi; se trovate un paesaggio che desiderate conservare potete prendere nota dei parametri necessari a generarlo o, ancora meglio, salvarlo su disco con l'opzione apposita, in modo da crearvi una vera e propria "collezione" di paesaggi.

Porgendo le nostre scuse ai lettori, riproponiamo il programma Boomerang, erroneamente saltato sul disco Amiga Magazine N. 13

Un gioco tutto nostro

(prima parte)

Per la rubrica Programmazione Facile di questo mese è stato scelto un argomento di sicuro interesse per un numero rilevante di lettori: come progettare e stendere le prime basi per la creazione di un videogioco utilizzando AmigaBasic.

Gianni Biagini

Scopo dell'articolo è l'analisi delle tecniche di gestione di Joystick, grafica e sprite
Per semplicità il tutto verrà sviluppato usando il linguaggio Basic, che pur non permettendo elevate velocità di esecuzione, risulta molto chiaro e comprensibile (va sottolineato che si ottengono risultati nettamente migliori compilando il programma una volta ultimato; il compilatore è un tipo di programma che permette di trasformare il codice sorgente Basic nell'equivalente in linguaggio macchina). Per realizzare un videogioco si deve sviluppare una trama: qui la fantasia può sbizzarrirsi nel creare le storie più diverse; chi non pensa già ad una bellissima principessa da salvare o ad una fuga alla ricerca della libertà? (in questo caso fracassare un muro di mattoni può essere uno dei tanti sistemi di evasione). Noi siamo futuristi, guardiamo all'avvenire, "Sia sgombra di mummie la soglia del futuro, largo ai giovani ai violenti, ai temerari", disse Marinetti, il creatore del movimento futurista nel suo famoso manifesto.
Spazio sia, allora; vogliamo occuparci subito dello scenario? Prendiamo un programma di grafica preferito e disegnamo un accattivante fondo stellare, magari con qualche immagine di pianeta qua e là. Partiamo dall'aspetto della programmazione: è stato creato uno sfondo che è stato salvato in formato IFF per farlo comparire sul nostro schermo, come possiamo fare? Insieme all'interprete Basic (sul disco Extras) vi sono una serie di esempi che mirano a rendere indipendenti i programmatori, uno di questi è LoadACBM. Che nome strano, cosa significa? Load dall'inglese vuol dire "carica" e ACBM è un formato grafico usato da Amiga (per la precisione è il formato non compresso) il cui nome deriva dall'acronimo inglese di Amiga Continuous BitMap. LoadACBM non fa altro che caricare da disco un'immagine e visualizzarla sullo schermo. Nessun programma salva in ACBM e questo potrebbe essere un guaio, ma basta sapere che sempre sullo stesso disco c'è un programma che cambia il formato da ILBM ad ACBM per rincuorarci, siamo a buon punto! Cosa fare allora per trasferire i due formati di cui sopra ?
Si disegna con il painter lo sfondo, lo si salva su di un disco e si lancia il programma LoadILBMsaveACBM che chiederà il nome del disegno da convertire. A questo punto bisogna inserire il dischetto contenente l'immagine e digitare DF0: NOMEDISEGNO dove df0: serve a dire ad Amiga di non cercare il file nella directory corrente, ma nel disco inserito, e NOMEDISEGNO rappresenta il nome del disegno al quale voi avete associato la vostra illustrazione. Dopo una serie di calcoli e di accessi allo schermo, il programma vi dirà di aver finito, la vostra schermata è ora salvata in formato ACBM e può essere letta via Basic dall'ormai famoso LoadACBM. Caricato lo sfondo dobbiamo preoccuparci di creare il gioco vero e proprio. In questa fase intervengono due elementi fondamentali: il joystick, che comanda lo svolgimento del gioco, e gli sprite, oggetti in movimento e questo significa che dobbiamo conoscerli a fondo. Riprendete il vostro Amiga Magazine di qualche mese fa, troverete tutto ciò che riguarda la gestio-

ne degli sprite dalla creazione all'utilizzo. Creiamo il nostro soggetto con il programma OBJEdit, sempre sul disco dell'AmigaBasic e salviamolo. Certo il nostro spirito ludico-decubertiano (che in parole povere significa che nei videogiochi *vogliamo dei nemici*) ci spinge a creare degli aggressori alla nostra navicella, a voi quindi lo sfogo di creare un mostro quanto più brutto possibile. Salvati i due sprite, siamo in procinto di creare la componente di dinamismo che caratterizzerà il nostro gioco. Avete mai visto sul manuale del Basic due comandi nominati rispettivamente STICK e STRIG? Ebbene essi sono delle istruzioni che permettono di leggere lo stato del joystick e del tasto fire. Incredibile, vero? Tutto si semplifica quindi, possiamo caricare uno scenario, possiamo definire uno sprite e muoverlo a piacere seguendo i movimenti del joystick; grande invenzione i linguaggi di programmazione !! Innanzitutto STICK è una funzione, questo implica che essa dà come risposta un valore e che questo valore può essere confrontato nel programma. Come sapere se l'utente sta muovendo il joystick verso destra? A livello di programmazione basta usare una delle strutture fondamentali di ogni linguaggio, la selezione. Scrivere IF STICK(2) = 1 THEN PRINT "Destra" significa chiedere al programma se sta muovendo la manopola a destra e in caso affermativo scrive sullo schermo Destra. Certo, provare questo piccolo listato può dare i suoi frutti, ma capire perchè è stato scritto in questo modo è più interessante. Chi mi dice, per esempio, che stick(2) controlla il joystick in orizzontale? La risposta è il manuale dell'AmigaBasic e, in secondo luogo io stesso. Stendiamo una piccola tabella per capire come leggere la direzione di spostamente e come distinguere i due joystick che può controllare Amiga.

```
STICK(0) direzione X del joystick A
STICK(1) direzione Y del joystick A
STICK(2) direzione X del joystick B
STICK(3) direzione Y del joystick B
```

Il valore ritornato dalla funzione STICK obbedisce alla tabella sottoindicata:

1 Movimente a destra o verso l'alto
0 Il joystick non è utilizzato
-1 Movimento a sinistra o verso il basso

E' chiaro che, a questo punto, è possibile scrivere un piccolo programma che riesca a leggere completamente le funzioni del Joystick: va creato un LOOP, ossia un ciclo di ripetizione, e all'interno dello stesso, far fare tutti i controlli del caso. Diamo un primo sguardo al programma seguente:

```
For N=1 to 100000
 IF STICK(2)=1 THEN PRINT "Destra"
 IF STICK(2)=-1THEN PRINT"Sinistra"
 IF STICK(3)=1 THEN PRINT "Alto"
 IF STICK(3)=-1 THEN PRINT "Basso"
NEXT N
```

Cosa fa? Conta fino a 100000 e, se durante il conteggio il Joystick viene mosso, scrive la direzione relativa. Geniale, ma in un videogioco spaziale è importante sparare, come facciamo leggere ad Amiga il tasto di fuoco ?
La soluzione si chiama STRIG ed è anch'essa una funzione (e quindi molto comoda per fare dei controlli sul valore che essa restituisce).
Come già fatto per STICK diamo la tabella dei valori corrispondenti alle mutue posizioni dei tasti di sparo.

STRIG(0) *restituisce il valore* 1 se il tasto fuoco del joystick 1 è stato premuto dall'ultima lettura del tasto stesso, altrimenti restituisce 0.
STRIG(1) restituisce 1 se il tasto fuoco del joystick 1 è premuto
STRIG(2) restituisce il valore 1 se il tasto fuoco del joystick 2 è stato premuto dall'ultima lettura del tasto stesso, altrimenti restituisce 0.
STRIG(3) restituisce 1 se il tasto fuoco del joystick 2 è premuto

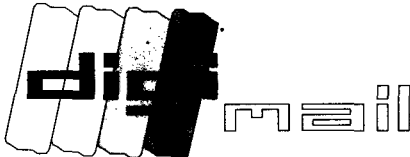
Il programma precedente potrà ora includere anche un ipotetico sparo, basta far precedere alla linea NEXT N l'istruzione:

```
IF STRIG(2)=1 THEN PRINT"Fuoco!!!"
```

Adesso è possibile finalmente utilizzare completamente il joystick: per determinare le direzioni di spostamento e l'eventuale pressione del tasto FIRE, è però necessario conoscere come creare il videogioco vero e proprio. Nel piccolo sommario in cui si specificavano i punti fondamentali da seguire: creare e caricare uno sfondo, creare e visualizzare gli sprite, e infine poterli muovere a libero arbitrio dell'utente. Sul numero 11 di Amiga Magazine è possibile leggere l'articolo di Programmazione Facile che dice tutto e anche di più sugli sprite. Se vi siete fatti sfuggire quel numero di Amiga Magazine, ripassiamo più veloci della luce (e siamo ancora in tema spaziale) quanto analizzato, tralasciando ovviamente l'aspetto di programmazione C che nel nostro caso non interessa (visto che lavoriamo in Basic) e vediamo come è possibile muovere uno sprite. Supposto di averlo creato con l'ObjEdit (il programma contenuto nel disco Extras fornito dalla Commodore (se non lo trovate cercate nella directory BasicDemos). A questo punto un minimo di trama è indispensabile: supponiamo di essere dei pianeti un po' pazzerelli che amano fare l'autoscontro nelle galassie. Se incontrate problemi nel creare lo sprite, sappiate che uno di questi oggetti è presente sul disco allegato, si chiama ball.spr e altro non è se non una pallina che potete comandare a vostro piacimento. Ricordando le preziose istruzioni di movimento per gli sprite, rivediamo il programmino sviluppato sopra: non vi sembra che invece di PRINT e della direzione si possa agire direttamente sulla posizione dello sprite ?
OBJECT.X numerosprite,ascissa dice ad Amiga di porre uno sprite alla locazione specificata da ascissa. Se avessimo una funzione che ci dice la posizione corrente dello sprite saremmo a cavallo perchè basterebbe sommarvi 1 per avere uno spostamento a destra o -1 per uno a sinistra. Questa funzione c'è, e si chiama OBJECT.X

(numerosprite). Vogliamo che spostando il joystick a destra lo sprite si muova nella direzione indicata? Niente di più facile, a questo punto, basta scrivere:

```
IF STICK(2)=1 THEN OBJECT.X
1,OBJECT.X(1)+1
```

Ripetiamo lo stesso discorso per tutte le altre possibili direzioni e avremo uno sprite comandato da noi. Attenzione però, ricordatevi sempre di scrivere OBJECT.ON alla fine degli IF relativi alle direzioni, così che il nostro pianetino possa essere posizionato ogni volta che avviene uno spostamento. Esistono, comunque, tecniche di movimento degli sprite più raffinate comunque la Commodore fornisce un manuale del Basic assieme alla macchina che può essere consultato per trovare qualsiasi soluzione.
Nasce ora un altro problema: il protagonista deve essere comandato via joystick, ma il nemico? Certo sarebbe bello che fosse lo stesso Amiga a decidere delle sorti della battaglia, ma come fare? La risposta è tutta nella casualità. Sorvolando volutamente il concetto statistico di casualità, viene istintivo pensare che se Amiga, che purtroppo non riesce a muovere il joystick potesse in qualche modo "pensare" delle mosse non prevedibili (questo è tipico per gli umani), potrebbe essere il nostro avversario. Il linguaggio Basic mette a disposizione una funzione il cui compito è quello di pensare a un numero compreso tra zero e uno pescando in uno spazio equiprobabile. Dobbiamo controllare quattro tipi diversi di movimenti? Ci basterà dividere in quattro parti uguali il nostro spazio e, sempre attraverso delle scelte, compiere un'azione coorrelata al numero ottenuto. Per essere più chiari, sapendo che RND(1) è la funzione che restituisce un numero pseudocasuale, potremo associare i numeri da 0 a 0,25 all'azione "vai a destra", quelli da 0,25 a 0,50 all'azione "vai a sinistra" e così via. In termini di linguaggio questo può scriversi come (ricordando che vogliamo muovere uno sprite a sinistra):

```
A=RND(1)
IF A>0.25 AND A<0.50 THEN OBJECT.X
1,OBJECT.X(1)-1
OBJECT.ON
```

Facciamo il punto della situazione! Abbiamo imparato a far "leggere" ad Amiga il joystick, abbiamo visto come muovere uno sprite a seconda di quanto comandato e come leggere da disco un'immagine creata con un painter (la nostra che forniamo è stata digitalizzata e rappresenta la terra in una foto scattata dall'Apollo 10 a 44000 km di distanza). Nel prossimo numero vedremo come unire questi moduli software per creare quello che sarà il prototipo del nostro videogioco, fermo restando che potrà essere da voi migliorato.
Sul prossimo numero di Amiga Magazine verranno analizzati una serie di "trucchetti" per velocizzare i giochini (questo include semplicissimi ragionamenti a livello statistico e ottimizzazioni del codice). ▲

Come utilizzare il dischetto

Prima di utilizzare il dischetto allegato alla rivista assicuratevi di averlo protetto dalla scrittura aprendone la finestrella, per evitare accidentali cancellazioni o contaminazioni da virus (il disco allegato è al 100% privo di virus ed il Gruppo Editoriale Jackson non si assume nessuna responsabilità in caso di contagio). A questo punto accendete il computer. Dopo la presentazione, dalla quale è possibile uscire premendo il tasto sinistro del mouse, verrà caricato il Workbench. Aprendo l'icona del disco (cliccando due volte con il tasto sinistro su di esso) appaiono i "cassetti" relativi a giochi, utilities, ecc. Aprite allo stesso modo il "cassetto" al quale siete interessati e caricate il programma desiderato. Alcuni programmi possono essere anche caricato da CLI con altre opzioni (vedere a parte le spiegazioni dei programmi). Per chi possiede Amiga con "soli" 512 K di memoria RAM è consigliabile chiudere tutte le finestre aperte tranne quella relativa al programma da far girare (che può essere eventualmente chiusa con un po' di prontezza di riflessi durante il caricamento). In tal modo il programma avrà a disposizione più memoria. In caso contrario, potrebbero verificarsi dei malfunzionamenti.
ATTENZIONE: prima di far partire ogni programma vi raccomandiamo di leggere attentamente le istruzioni del programma. I programmi Xboot, Amicron, Spline, Surf, Scenery sono di pubblico dominio, e possono essere liberamente distribuiti. Inoltre, ricordiamo che, i programmi sorgenti ed eseguibili delle pagine di TRANSACTOR per AMIGA, si trovano nel cassetto TRANSACTOR e devono essere caricati da CLI. Tutti gli altri programmi presenti sul dischetto sono, invece, coperti da Copyright 1990 Gruppo Editoriale Jackson-Compute! Publications Inc., e come tali non devono essere copiati e distribuiti senza la preventiva autorizzazione scritta dell'editore. Amiga Workbench 1.3 è copyright 1985, 1986, 1987, 1988, 1989 Commodore-Amiga Inc., tutti i diritti sono riservati.
Per ogni questione tecnica riguardante il funzionamento del disco telefonare il Martedì e il Giovedì dalle ore 14.30 alle ore 16.00 al 02-6948287.

Come entrare in CLI

- Accendere il computer
- Inserire il disco con il Workbench
- Attendere il caricamento
- Cliccare due volte sul disco del Workbench
- Cliccare due volte sull'icona SHELL (per il nuovo CLI), oppure aprire il cassetto System
- Selezionare l'icona CLI

Esempi di caricamento da CLI
- Entrare in CLI
- Digitare:
COPY FROM C: TO RAM:
PATH RAM :
- Inserire il disco di Amiga Magazine e digitare CD DF0:
- Digitare CD UTILITIES

A questo punto, per esempio, si può digitare:

- Xboot

Amiga Magazine n. 14

SERVIZIO LETTORI **Compilare e spedire in busta chiusa a: GRUPPO EDITORIALE JACKSON Area Consumer - Via Pola, 9 - 20124 Milano**

A) Come giudichi questo numero di Amiga Magazine ?

- ❑ Ottimo
- ❑ Molto Buono
- ❑ Buono
- ❑ Discreto
- ❑ Sufficiente
- ❑ Insufficiente

B) Quale (i) articolo (i) o rubrica hai apprezzato di più?

Quale meno?

C) Cosa ti piacerebbe leggere nei prossimi numeri di Amiga Magazine ?

D) Ti è piaciuto il Software ?

E) Quante persone leggono la tua copia di Amiga Magazine ?

F) Possiedi un computer? Quale?

Quale (i) computer intendi acquistare in futuro?

G) Leggi altre riviste Jackson?
❑ SI ❑ NO

Quali?

H) Leggi altre riviste del settore?
❑ SI ❑ NO

Quali?

I) Oltre alle riviste dedicate al computer quali sono le tue letture preferite?

L) Quali sono i tuoi hobbies e maggiori interessi?

- ❑ Sport
- ❑ Musica
- ❑ Videoregistrazione
- ❑ Hi - Fi
- ❑ Fotografia
- ❑ Automobile
- ❑ Moto
- ❑ Viaggi

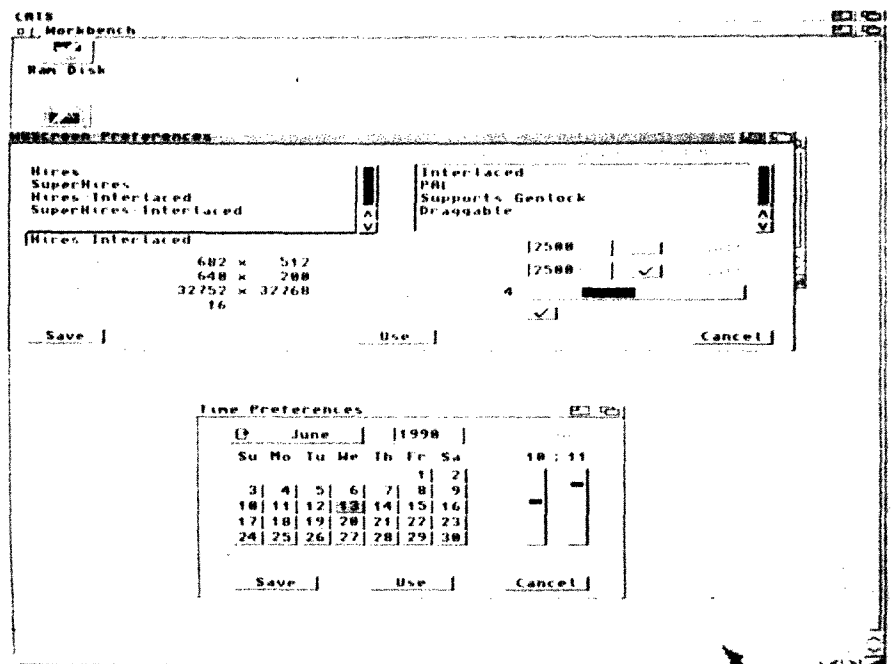
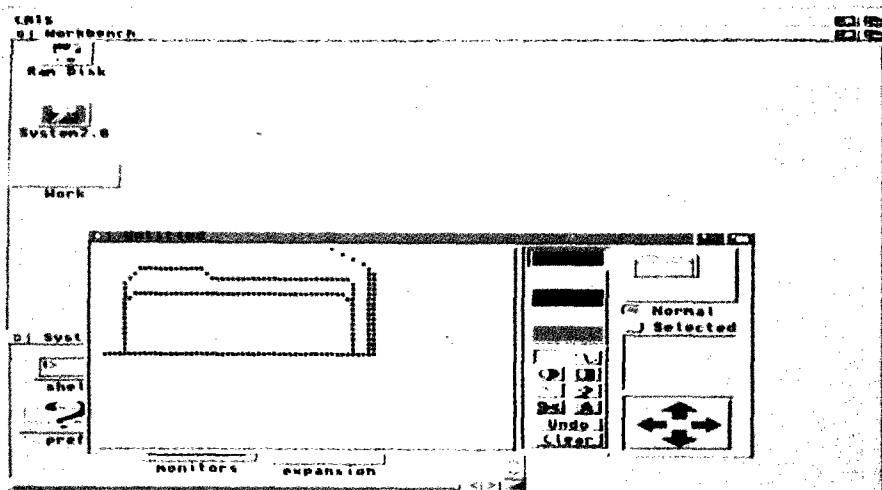
Nome
Cognome
Indirizzo
Età **Professione**
Città
Prov. **C.a.p.** **Tel.**

2.0: il futuro di Amiga

Sebastiano Vigna

Dopo tanto tempo, tante indiscrezioni, tante speranze, eccoci finalmente a parlare della nuova versione del sistema operativo di Amiga.
Com'è noto, la Commodore ha messo veramente molta carne al fuoco, e in un certo senso il destino di Amiga è legato al successo o meno di questo prodotto. La materia da trattare è tale e tanta da rendere pressoché impossibile un'esposizione uniforme, e quindi cercherò di dare un'idea delle novità in ciascuno dei settori più importanti, lasciando l'approfondimento ad un momento successivo.
La prima sorpresa è il nome: probabilmente per ragioni d'immagine, la Commodore ha deciso di chiamare la nuova versione 2.0 anziché 1.4. Da un punto di vista di ingenierizzazione del software, la cosa è più che giustificata (il lavoro svolto meritava certo un incremento maggiore di 0.1), ma il fatto che la decisione sia saltata fuori solo all'ultimo momento lascia un po' da pensare.

Il Workbench

La tanto discussa interfaccia grafica di Amiga ha subito una (fin troppo?) grossa serie di rimaneggiamenti. Innanzitutto, il look ora è diventato "tridimensionale", o, se vogliamo, "NeXToso". Adeguandosi a quelli che stanno diventando i canoni universali in fatto di GUI (Graphic User Interface), la Commodore ha ottenuto un risultato estetico veramente eccezionale, a scapito del completo disorientamento in cui cadranno tutti gli Amigoidi, da tempo abituati, nel male e nel bene, ai gadget di Intuition. Un esempio? I due gadget rettangolari in alto a destra, una volta deputati a mandare avanti o indietro le finestre, svolgono ora un ruolo diverso: il primo fa da zoom (in/out) il secondo manda alternativamente avanti/indietro la finestra a cui appartiene. La ragione di questa stranissima filosofia è veramente semplice: la compatibilità. Aggiungere un gadget avrebbe richiesto una disposizione della finestra leggermente differente, e questo avrebbe potuto condurre a strane interazioni tra programmi e sistema operativo. Per la stessa ragione anche sotto 2.0 non avremo gadget quadrati se non in 640 x 200.
Il look non è però solamente curato, ma anche incredibilmente flessibile: lo sfondo di ogni finestra del Workbench può essere riempito con un pattern differente; menu, titoli e icone sono disegnati con tre font selezionabili separatamente in modo arbitrario (ad esempio, l'Helvetica dà la netta impressione di essere sotto OS/2...). La finestra del Workbench contenente le icone dei dischetti, prima esclusivamente di tipo "backdrop" (cioè inchiodata allo sfondo, tanto da non sembrare neanche una finestra) ora può diventare "normale", vale a dire dimensionabile e spostabile; e, cosa assolutamente fondamentale, oltre ad aver cambiato tutti i menu del Workbench, la Commodore ne ha aggiunto uno nuovo che svolge un ruolo analogo al menu mela del Macintosh, permettendo ad applicazioni esterne (in genere "piccole") di integrarsi con il sistema operativo per estenderne le funzionalità.
Ma la novità forse più eclatante è il

Workbench asincrono! Dietro questa parola apparentemente misteriosa c'è una realtà molto semplice: le operazioni non vengono più gestite in modo monotask. Il risultato più evidente è che si può aprire un disco e lanciare un'icona ivi contenuta prima che tutte le icone del disco siano state lette. Inoltre, se per qualche ragione una finestra viene scoperta mentre il Workbench sta eseguendo qualche operazione (ad esempio, lanciando un programma) le icone riappaiono mentre l'operazione viene effettuata, e non alla fine.
Sembrerebbe proprio che il Workbench possa diventare qualcosa di realmente utilizzabile, ma resta sempre il fatto che solo una piccola parte dei file presenti su un disco è accessibile... o meglio era accessibile, dato che ora è possibile dire al sistema operativo di assegnare un'icona di default a tutti i file e directory presenti sul disco, e non solo a quelli dotati di un file ".info". Se a questo si aggiunge la possibilità di visualizzare i file in forma alfabetica (come da CLI) anziché in forma iconica, ci si rende conto che il Workbench può ormai soppiantare parecchie utility di gestione file.
Siamo ormai al livello di praticità del Finder (se non oltre), e da un punto di vista globale siamo anche in una situazione molto più flessibile, dato che la CLI ci permette anche un accesso "tradizionale".
A proposito di CLI: la Shell è finalmente in ROM (era anche ora...) e sono disponibili delle funzioni di cut/paste da una finestra all'altra.
Parecchi comandi, inoltre, sono diventati built-in, e cioè non dovranno più essere caricati da disco.
Grandi cambiamenti anche in zona Preferences: il caro vecchio programmino si è ormai esteso ad un flessibile e completamente riconfigurabile editor, a cui si potranno agganciare dinamicamente i programmi di configurazione di terze parti. Vale a dire, attraverso le preferences potremo in un futuro non lontano selezionare il comportamento di default di tutti i programmi che usiamo, e non solo del Workbench.

Exec

Come descrivere in poche righe quello che è successo? Forse basterebbe dire che le ROM passeranno a 512K. Il raddoppio dello spazio necessario al contenimento del codice parla già della rivoluzione avvenuta in seno al sistema operativo, rivoluzione rispetto alla quale la nuova interfaccia grafica e il nuovo Workbench non sono che la punta di un iceberg.
Prima di tutto, un enorme numero di funzionalità è stato aggiunto a tutti i livelli. Per chi conosce ARP, pressoché tutte le sue funzioni (con i nomi modificati, per evitare confusioni e pericolose sovrapposizioni) sono andate a finire nelle nuove librerie. Alcune sono state addirittura migliorate: ad esempio GADS, che esegue la suddivisione di una linea di comandi in argomenti numerati opportunamente, può ora convertire direttamente un argomento da stringa ASCII in numero intero, se richiesto. Tutte le mancanze gravi sono state sistemate, e, in previsione delle numerose schede di rete che stanno uscendo, è stato implementato un meccanismo di record locking.
A livello molto più "basso", Exec ora supporta pienamente il 68030 e il 68020, con tutte le complicazioni derivanti dall'uso delle data/memory cache.
Dato che i chip custom di Amiga possono scrivere in una zona di memoria indipendentemente dalla CPU, può infatti succedere che i dati immagazzinati dal 680x0 nelle cache memory non corrispondano agli effettivi contenuti della memoria.
Una novità da sempre attesa è il controllo dello stack: ora direttamente a livello di Exec è possibile richiedere di controllare se un task deborda dallo spazio assegnatogli. Il controllo viene eseguito scrivendo una parola "improbabile" in fondo allo stack, e controllando periodicamente se la parola non è stata cancellata (ovviamente il task potrebbe per caso debordare ma riscrivere esattamente la stessa parola: il problema è evitato scegliendo, appunto, un codice "improbabile"). I meccanismi di configurazione all'accensione sono stati raffinati (ad esempio, prima la device audio veniva sempre attivata, anche se nessuno la usava, e consumava memoria per niente) e, soprattutto, alcune zone di uso frequente (lo stack supervisore, ad esempio) che prima spendevano il loro tempo peccaminosamente in Chip RAM (50% in meno di velocità di accesso) sono passate a miglior vita in Fast RAM.

Intuition

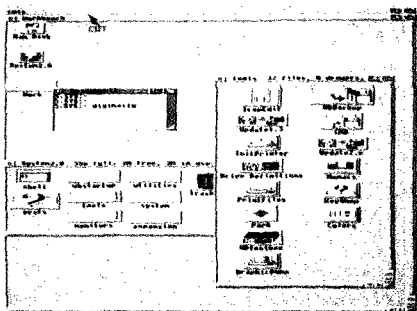
Alcune funzioni rendono ora più semplice la creazione di menu indipendenti dai font utilizzati, cosa necessaria in un ambiente dove il

font di sistema è completamente arbitrario, e una nuova funzione, ChangeWindow, provvede a muovere e dimensionare una finestra con una sola azione (prima si era costretti a eseguire una SizeWindow() e una MoveWindow(): essendo però queste funzioni asincrone, eseguirle consecutivamente diventava una vera piaga). L'introduzione della ASL.library, inoltre, porrà fine al dilagare di decine di file requester diversi, dato che consentirà di accedere a font e a file in maniera completamente standard, attraverso una semplice chiamata di sistema (come del resto avviene da sempre su GEM, su Macintosh e su Windows); la possibilità di gadget aggiuntivi e di callback di funzioni ne renderà la gestione infinitamente flessibile.
Il concetto di schermo ha ora una nuova dimensione: le specifiche fisiche (ad esempio, 640 x 512) sono indipendenti da quelle logiche (fino a 32760 x 32768). Se ci si sposta contro al bordo del video, il contenuto "scrolla" in modo perfettamente liscio verso l'alto, scoprendo un'altra zona di schermo! Sono anche stati implementati degli schermi "pubblici", sui quali più programmi si possono dare appuntamento per lavorare insieme. Esempio: il vostro editor preferito e il vostro programma di DTP. Dulcis in fundo, l'overscan è ora legalmente supportato dal sistema operativo, e quindi è passato il tempo dei sotterfugi di programmazione più contorti a cui eravamo abituati sotto 1.3. Le possibilità aperte sono veramente illimitate.
I font, finalmente, saranno scalabili, vale a dire se richiedete il Times 24 ma disponete solo del Times 18, il sistema operativo "ingrandirà" automaticamente quello disponibile, con il prevedibile effetto a "pallettoni" ma mettendo in una situazione più umana l'utente medio. Si prevede comunque, qualche tempo dopo l'introduzione della 2.0, di dotare Amiga della tecnologia outline Intellifont della Agfa Compugrafic, il che significa font perfetti a tutte le dimensioni e su qualunque periferica (questa tecnologia è per ora utilizzata solo da Page Professional). L'output di testo, inoltre, è stato ottimizzato, e il supporto per i ColorFont è stato inserito nelle ROM.

AmigaDOS

Ho voluto però tenere in fondo la notizia migliore: AmigaDOS è stato completamente riscritto in C, e il Fast File System è in ROM. Questo, come abbiamo già detto tante volte, significherà moltissimo per tutti i programmi file-oriented (database per esempio) perchè, sebbene l'hardware dei dischi impedisca di ottenere una velocità "grezza" di lettura/scrittura molto più alta di quella corrente, lo spaventoso affaticamento del sistema in corrispondenza ad ogni chiamata AmigaDOS verrà finalmente eliminato (avete mai provato a leggere un file RAM di 10K un byte alla volta?). E le funzioni aggiuntive presenti a tutti i livelli renderanno un vero piacere scrivere programmi per Amiga! Cito a caso: hard links, notificazione automatica di modifiche, I/O bufferizzato... sembra la lista dei desideri dello sviluppatore Amiga medio.
L'inclusione di ARexx nel software di sistema, inoltre, consentirà di bypassare il poco flessibile script language di AmigaDOS, e di scrivere finalmente dei batch file degni di questo nome.

Conclusioni

E' proprio quello che tutti aspettavamo: un vero, potente, versatile, attraente sistema operativo per il nostro Amiga. Se la Commodore continuerà a viaggiare in questa direzione, non ci si può aspettare che bene.
L'unica critica che mi sento di fare è estetica, vale a dire i gadget non quadrati. Non credo che le ragioni di compatibilità possano essere così pressanti quando l'utente può scegliere senza colpo ferire un font alto 24 punti per il titolo di una finestra. Andy Finkel non era di questo avviso al DevCon di Parigi, e senza dubbio la sua parola vale più di quella di chiunque altro; ciononostante, dover vivere con l'incubo delle icone schiacciate non è esattamente il massimo.
La prima preoccupazione di base è la compatibilità: data la mega-riscrittura del sistema operativo, è lecito aspettarsi il breakdown di una buona fetta di software. Non va però dimenticato che le release beta sono da parecchio tempo in mano ai produttori di software, che dovrebbero quindi essere ormai adeguati. Ma il vero problema, purtroppo, è un altro: dato che l'uscita pubblica negli USA è pensata attorno a Settembre, non credo di essere pessimista dicendo che ancora all'inizio del '91 ci saranno molti Amigoidi con la 1.3.
Quante software house utilizzeranno nei loro prodotti le nuove chicche della 2.0 sapendo che una così grossa fetta di utenza si vedrebbe impossibilitata ad usarli (e quindi a comprarli)? Il dilemma, emerso tra l'altro al DevCon, non è affatto semplice da risolvere.
Una possibilità è il mantenimento, per un certo periodo dil tempo, di versioni parallele, ma il costo e la fatica dissuaderanno, a mio parere, le software house da una politica di questo tipo. E ciò significa che dovremo aspettare parecchio perché la comunità Amiga "passi" veramente alla 2.0: per i primi mesi, i vantaggi saranno essenzialmente dati dalla nuova interfaccia e dal nuovo Workbench. Ma dato che, come abbiamo visto, si tratta di vantaggi veramente grandiosi, non posso che consigliarvi di comprare al volo il kit di upgrade per la nuova versione non appena uscirà.
Speriamo presto. ▲

The last Amiga

In attesa di una completa prova su strada, diamo un'idea dei principi che stanno sotto il nuovo gioiello Commodore.

Sebastiano Vigna

L'uscita del 3000 in concomitanza alla 2.0 (già non si chiama più 1.4...) è l'avvenimento dell'anno. Il nuovo modello della Commodore, unitamente alla nuova versione del sistema operativo, spinge in primo piano, e in maniera prepotente, i modelli della serie Amiga, fino ad oggi riservati ad un gruppo di appassionati, ma sconosciuti al largo pubblico: ne è testimone la copertina di BYTE, dedicata proprio al 3000.

32 bit

La filosofia dominante di tutta la progettazione è: largo ai 32 bit, quelli veri, implementati dal processore, agli slot, alla memoria.
Un poderoso 68030 (a 16 o 25 MHz, ci sono due versioni) coadiuvato da una fedele FPU (68881 o 68882, sempre a seconda delle due versioni) mette il 3000 in grado di rivaleggiare con un mini quanto a potenza di elaborazione.
La memoria di base (2 Mega) è espandibile internamente a quattro Mega, ma tramite il bus a 32 bit è possibile indirizzare la bellezza di 4 Gigabyte.
Un nuovo standard a 100 pin per gli slot (che sono quattro), denominato Zorro III (quello del 2000 si chiama Zorro II) permette di accedere integralmente alle risorse del sistema, pur rimanendo compatibile con lo Zorro II; uno slot video e uno per eventuali schede CPU (68040...) completano il sistema. Ci sono anche due slot di tipo PC/AT, in modo da permettere l'uso di schede bridgeboard.

Il 3000 arriva, inoltre, dotato di hard disk (40 o 100 Mega) e di una porta SCSI che permette di aggiungerne a piacere ad un costo ridotto.
Sfruttando tutti i 32 bit del bus interno, il controller raggiunge velocità veramente iperboliche.
Altra novità di serie è una scheda, detta dalla Commodore Display Enhancer, che in modo totalmente analogo al Flicker Fixer della Microway provvede a deinterlacciare il segnale video se necessario.
Si noti, se necessario: contrariamente al Flicker Fixer, i modi non interlacciati non vengono sottoposti alla bufferizzazione, che darebbe uno spiacevole effetto di "scia" durante le animazioni. Inoltre, il Display Enhancer è in grado di gestire il PAL in modo completo (overscan, etc.).

L'ECS

Nel 3000 fa per la prima volta ufficialmente la sua comparsa il tanto mormorato ECS (Enhanced Chip Set), ovvero le nuove versioni dei celebri chip custom di Amiga Agnus, Paula e Denise.
La nuova versione di Agnus è già correntemente montata sui 2000 dalla 6.0 in su, ma nel 3000 viene montato un modello ancora più potente, capace di indirizzare 2 Mega di Chip RAM.
Le dimensioni massime di manipolazione del blitter sono invece le stesse, vale a dire 32768 x 32768 (la vecchia versione maneggiava solo 1024 x 1024 bit).
Gli altri due chip provvedono a produrre i nuovi modi grafici, le cui possibilità combinatorie hanno ormai raggiunto i livelli di guardia, perché ogni nuova opzione si può applicare a tutti o quasi i modi grafici precedenti.
Gli effetti più salienti sono il cosiddetto "productivity mode", che dà 640 x 480 pixel senza interlacciamento, ma con una palette di due o quattro colori su 64, e il 1280 x 512 con interlacciamento (per titolatrici video e simili).
Il productivity mode è comunque utilizzabile solo da chi possiede un monitor MultiSync, in quanto la fre-

Caratteristiche tecniche di Amiga 3000

Microprocessore:
- Motorola 68030 (32 bit) e co-processore aritmetico 68881 da 16 MHz, oppure 68030 e co-processore 68882 da 25 MHz

Memoria:
- 1 Mbyte Fast RAM, estendibile a 2 Mbyte sulla scheda madre
- 1 Mbyte memoria di sistema, estendibile a 4 Mbyte sulla scheda madre (oppure a 16 Mbyte con chip da 4 Mbit)

Adattatore di interlacciamento video standard

Sistema rapido di controllo del disco rigido SCSI

Orologio interno in tempo reale con back-up a batteria

Connettori di espansione:
- 4 connettori di espansione Zorro III a 32 bit (compatibili con i connettori A2000)
- con 2 connettori d'espansione PC AT
- 1 connettore d'espansione video
- 1 connettore per bus locale 68030, 32 bit, 200 piedini, per memoria cache e co-processori

Interfacce:
- Connettore a 23 piedini video Amiga (15,75 kHz)
- Connettore a 15 piedini (formato VGA-31,5 kHz)
- Uscita audio stereo
- Connettore esterno SCSI a 25 piedini
- Connettore interno SCAI a 50 piedini
- Connettori serie, parallelo, disco esterno
- Connettore per tastiera
- 2 connettori per mouse/joystick/ penna ottica

BUS di sistema:
- Accesso da 32 bit alla memoria di sistema
- Accesso a 32 bit alla memoria video
- Circuito integrato specializzato, in grado di generare i chip di memoria nei modi "colonna statica" e "burst" del 68030
- Controllore SCSI specializzato, per generare l'accesso diretto alla memoria (DMA) a 32 bit

Unità centrale:
- Unità centrale compatta, capace di accogliere 3 lettori interni da 3,5" (2 dei quali accessibili dall'esterno).

Compatibilità:
- con i sistemi e le periferiche Amiga esistenti (per 500, 2000 e 2500)
- con l'ambiente MS-DOS, grazie alla tecnologia a schede modulari (bridgeboard)

aspettarci che tutti lo comprino perché è meraviglioso e non costa niente: deve anche funzionare bene, e disporre di una larga fetta di software applicativo altrettanto ben funzionante. La Commodore ha capito molto bene questo fatto: il rilascio del sistema di Authoring AmigaVision in contemporanea al 3000 ad un prezzo molto basso e la presentazione delle schede per la messa in rete, lo mostrano chiaramente. D'altra parte, non stiamo parlando di un computer per tutti: la versione più piccola ha un prezzo ufficiale di 5.500.000 lire (IVA esclusa), ed è già destinata ad un'utenza dotata di fondi. Nessuno spende parecchi milioni per giocare ai videogame, ma è nelle speranze della Commodore (e nelle nostre) che 3000 conquisti molti nuovi utenti. ▲

quenza di rinfresco è più alta del PAL. Anche a questo scopo, sul retro del 3000 troviamo un connettore VGA. Un aspetto estremamente importante ma spesso poco sottolineato dei nuovi chip è la loro capacità di passare dal modo NTSC a quello PAL via software. Questo significherà una maggior facilità nel debug di applicazioni destinati ai due mercati, dato che, ad esempio, le case americane non dovranno più procurarsi un Amiga PAL apposta per testare i loro prodotti. Anche le capacità di genlock sono state migliorate, ed ora permettono una gestione molto più libera dei bordi trasparenti e, soprattutto, di rendere trasparente qualunque colore della palette, e non solo quello di sfondo (il colore 0), come in precedenza.
Tutto questo (e anche di più) è contenuto in uno chassis di dimensioni veramente ridottissime. Il risultato è stato ottenuto grazie a cinque nuovi chip speciali (custom gate array) nei quali è stata concentrata buona parte della logica di controllo. Il fatto di avere meno spazio non ha comunque impedito alla Commodore di dotare il 3000 di un'abbondante serie di connettori: seriale, parallelo, SCSI, per floppy esterni, video Amiga, VGA, e audio. Il pulsante di accensione, ora, si trova sul fronte, mentre i connettori per mouse e tastiera sono (fortunatamente) passati di lato. Esteticamente, un bel passo avanti.

Conclusioni

Che dire di fronte ad una macchina come Amiga 3000? Abbiamo fatto un bel passo in avanti, la velocità è strabiliante, e ci siamo finalmente lasciati dietro il flickering.
Ma dietro l'ottimismo che porta sempre con sé un nuovo prodotto, dobbiamo essere realisti: anche il più misero compatibile MS-DOS con una scheda VGA può visualizzare 256 colori indipendenti. E' vero, abbiamo l' HAM, ma ciò non toglie che 256 colori indipendenti su 16 milioni siano ormai un minimo sotto il quale non si può andare senza tanti complimenti. E in alta risoluzione, siamo ancora a livello 16. Se la Commodore non affronterà, e in fretta, questo problema, non ci saranno speranze per la serie Amiga in campo desktop video. Voglio dire che il 3000 è una macchina a "livello giusto" per la tecnologia attuale, ma non è un prodotto altamente innovativo al livello del 1000 ai tempi della sua uscita. Quindi non possiamo

Amiga 3000
sarà proposto in tre modelli:

- Commodore AMIGA 3000-16-40 con processore Motorola16 MHz 68030 e 68881, 1 Mb di Chip RAM, 1 Mb di Fast RAM e hard disk da 40 Mbyte 19 ms Lire 5.500.000 + IVA
- Commodore AMIGA 3000-25-40 con processore Motorola25 MHz 68030 e 68882, 1 Mb di Chip RAM, 1 Mb di Fast RAM e hard disk da 40 Mbyte 19 ms Lire 6.500.000 + IVA
- Commodore AMIGA 3000-25-100 con processore Motorola25 MHz 68030 e 68882, 1 Mb di Chip RAM, 1 Mb di Fast RAM e hard disk da 100 Mbyte 19 ms Lire 7.500.000 + IVA

Grazie Commodore !!

Il nuovo Amiga 3000, AmigaVision e altre varie considerazioni

Massimiliano Anticoli

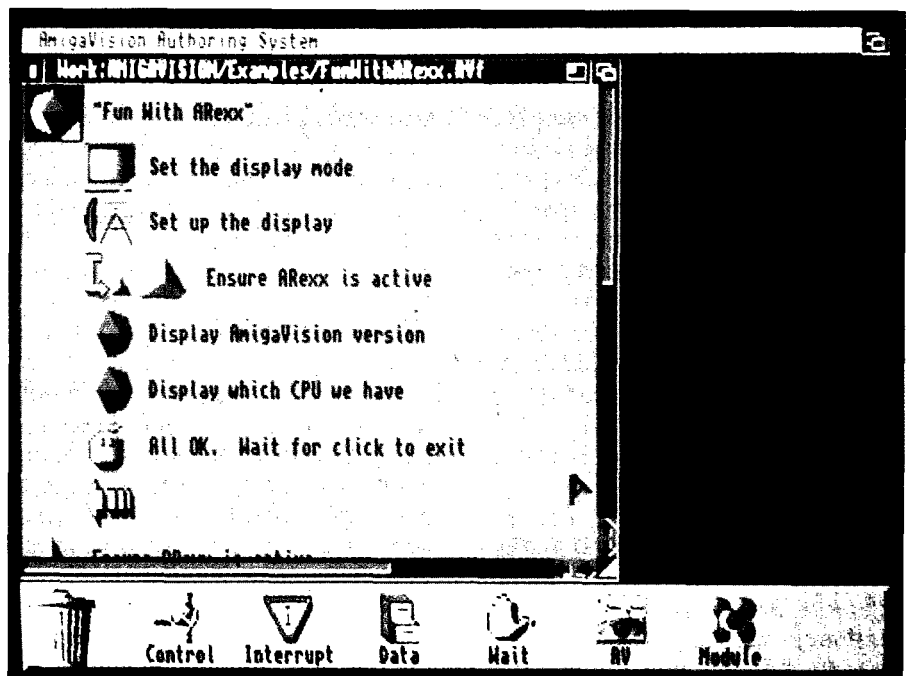
Dopo tante voci di corridoio, ecco il fatidico comunicato stampa sulla mia scrivania: il nuovo Amiga 3000 è nato !
In Italia, il 24 Aprile, non c'è stata una presentazione ufficiale come al Lincon Center di Manhattan o al Sicob/ Comdex di Parigi, ma circa un mese più tardi è arrivato, in redazione, un invito per andare a "gustare" il nuovo gioiello Commodore.

In Commodore

Quando l'ho visto non credevo ai miei occhi!
Veramente stupendo questo 3000 (anche il look rinnovato), per non dire del Workbench: talmente meraviglioso da far impallidire OS/2 e Windows 3.0 !
Il 3000, come ha spiegato Werter Mambelli - Amministratore Delegato della Commodore Italiana, è un computer orientato al mercato professionale multimediale, poiché ha prestazioni non riscontrabili in nessun altro sistema della stessa fascia di prezzo. Peccato! Perché, a mio modesto parere, il 3000 è un computer che può essere lanciato in qualsiasi mercato, costatando la struttura interna (68030, ecc.) e non vedendo concorrenti (si, anche i vari PC con 80486 Intel e i Mac).

Multimedialità

Moltissime riviste prestigiose, tra cui Byte e Bit, hanno dedicato interi speciali a questo argomento.
Un'applicazione denominata "multimedia" deve gestire e integrare almeno tre elementi fondamentali:

Una schermata del programma Amiga Vision

testi, grafica, animazioni, suono, sintesi vocale e video.
Una stazione di lavore "multimediale" deve permettere l'interfacciamento ai numerosi dispositivi come: CD-ROM, telecamere, digitalizzatori, videoregistratori ecc.
Naturalmente il dispositivo di output principe sarà il monitor di Amiga.
Una workstation multimediale può essere utilizzata per: business presentation, addestramento con computer, produzione spot televisivi e affini, simulazione, post-produzione e, per il futuro non tanto lontano, dati video fondamentali, utile, ad esempio, ad agenzie immobiliari (la casa presentata vista da computer) e medici (cartella cliente completa), ecc. Le applicazioni multimedia sono solo limitate dalla fantasia dell'utente !

AmigaVision

Dato che Amiga 3000 è stato lanciato come computer multimediale (notate le caratteristiche tecniche non c'è nulla da eccepire !) c'è bisogno di un programma che permetta di allestire facilmente, ad esempio, una presentazione combinando la interconnessione tra immagini fisse, immagini video, testi, grafica, suono e sintesi vocale.
Il programma in dotazione con Amiga 3000 si chiama AmigaVision.
Quest'ultimo si basa su una intuitiva interfaccia utente basata su icone, dunque non è necessario conoscere alcun linguaggio di programmazione, ed è sufficiente spostare le suddette icone per creare un diagramma di flusso in modo da creare collegamenti e informazioni; naturalmente AmigaVision sfrutta le possibilità multitasking di Amiga, cioè di poter gestire contemporaneamente diversi programmi e quindi diverse funzioni.

Disponibilità

I nuovi Amiga 3000 saranno disponibili da Settembre, assieme al WB 2.0 e AmigaVision.
Per fine anno, inizio '91, tutti i 2000 avranno di serie l'ECS, il WB 2.0 e AmigaVision, per queste date saranno inoltre disponibili gli up-grade per i 2000 già venduti.
E per il 500 ?
E' disponibile solo AmigaVision ma è necessario almeno 1 Megabyte di memoria, mentre gli up-grade per il WB 2.0 e l'ECS per ora non ci sono, e non solo in Italia !
Quest'ultima notizia mi ha un po' sconcertato, visto il grandissimo budget pubblicitario investito per questa graziosa macchina, poiché la struttura interna di base del 500 è quantomai simile a quella del 2000. Speriamo che la Commodore si accorga di questa piccola mancanza, almeno per non far proliferare il mercato "nero" degli up-grade selvaggi.

Compatibilità

I programmi scritti precedentemente per i 500 e i 2000, girano perfettamente, o quasi.

Il futuro del 3000

Considerando i vari accorgimenti tecnici, non pochi, come, ad esempio, collegarsi in rete ecc, sono sicuro di un esaltante successo, visto anche gli allettanti prezzi !, e auguro ogni bene a questa favolosa macchina ! Sicuramente, d'accordo con Mr.Thompson (Amiga Magazine n.12) non mi resta che dire: Amiga 3000 in ogni ufficio e, aggiungo, in ogni casa !

La nuova struttura Commodore

La Commodore, come ci ha spiegato Werter Mambelli, ha diviso in due aree i suoi prodotti.
Nella fascia alta, chiamata sistemi, ci saranno i vari PC Compatibili e gli Amiga 2000/3000, mentre nella fascia consumer saranno compresi i computer a 8 bit e Amiga 500.
Inoltre, i sistemi di fascia alta, dunque anche il 3000, saranno venduti da concessionari autorizzati (Commodore Point con una certa professionalità, provata da Commodore, e alcuni nuovi), per la fine di quest'anno la Commodore prevede ad arrivare a 80; infine saranno potenziati i centri di assistenza (non solo per i computer di fascia alta).

Noi di Amiga Magazine

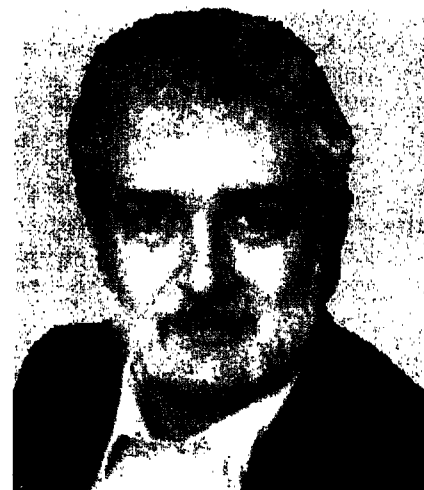
Promuoviamo a pieni voti: Amiga 3000, Workbench 2.0, AmigaVision e la Commodore Italiana, con in testa il simpaticissimo Amministratore Delegato Werter Mambelli.
Rimandiamo: la decisione della Commodore, di non rendere disponibile, per il momento (e queste due righe servono a tale scopo), l'upgrade per il 500. Non mi resta che concludere ricordandovi che nel numero 16, cioè quella del mese di Ottobre, vi proporremo una prova su strada di Amiga 3000, WB 2.0 e AmigaVision. ▲

Appunti, font, template e altro

Jim Butterfield

Questa volta ci occuperemo di alcuni semplici trucchi per il CLI. In primo luogo, potete pulire uno schermo pieno, premendo i tasti ESC e C (se usate la Shell, dovete premere anche RETURN). A questo modo si eliminano anche i modi speciali, come il set alternato dei caratteri, che viene impostato quando premete CTRL-N in una finestra CLI (non in una Shell).
Se volete lasciare un appunto temporaneo in una finestra CLI, premete il tasto del punto e virgola (;) e poi digitate l'appunto. Se non volete che il vostro appunto scorra via dallo schermo, create una nuova finestra digitando NEWCLI o NEWSHELL. Selezionate la nuova finestra, ridimensionatela, se lo desiderate, e digitate là il vostro appunto (non dimenticate di usare il tasto del punto e virgola). Selezionate di nuovo la finestra CLI principale e l'appunto resterà visibile finché non ritornate alla finestra precedente e digitate ENDCLI o ENDSHELL.
Un appunto più duraturo potrebbe essere creato con il breve comando COPY * TO RAM:APPUNTO. L'asterisco indica la finestra corrente, quella in cui il comando è stato digitato. Digitate il vostro messaggio, inserendo quante linee desiderate. Quando avete finito premete CTRL-BACKSLASH (\) per concludere il messaggio, e a questo punto ritornerà il prompt del CLI. Per vedere l'appunto, digitate TYPE RAM: APPUNTO.
Per usare il vostro Amiga come una macchina da scrivere, digitate COPY * TO PRT: e poi scrivete il vostro messaggio, terminando con CTRL-BACKSLASH.
Se desiderate lanciare un programma Amiga Basic, potete digitare AMIGABASIC e poi caricare il programma selezionato quando il Basic è pronto. Ma è più veloce far seguire il nome del programma al comando CLI. Così per lanciare un programma Amiga Basic chiamato "Grafici", digiterete AMIGABASIC GRAFICI.
Questa tecnica è utile specialmente per preparare dischi di boot personalizzati.
Aggiungete un comando simile a quello mostrato qui sopra al file S:Startup-Sequence e il programma BASIC scelto verrà lanciato quando farete il boot del sistema. A proposito, i programmi Amiga Basic possono scomparire dallo schermo quando terminano; tutto ciò che occorre è il comando SYSTEM alla fine del vostro programma.

Ridirezione

Un comando come COPY vi chiede di specificare l'input e l'output; non esistono valori di default, dal momento che i nomi li dovete fornire voi. Altri programmi o comandi possono avere dei valori di default: l'output, in particolare, potrebbe essere diretto verso la finestra corrente. Questo accade con comandi come Dir, List e Type; il risultato viene diretto verso lo schermo a meno che non venga ridiretto. Per ridirigere l'output di un comando o di un programma, usate il simbolo maggiore-di seguito dal nome di file o di device verso il quale volete che si diriga l'output. Supponiamo che vogliate inviare alla stampante l'elenco dei file contenuti in df1:. Digitate DIR >PRT: DF1:. Potete mettere uno spazio dopo il simbolo maggiore-di, ma ricordate che una ridirezione come >PRT:, con o senza uno spazio, deve seguire immediatamente il nome del programma [ed essere separato da quest'ultimo da uno spazio, N.d.T.].

Perché è importante la posizione? Qualche volta c'è più di un comando e viene generato più di un output. Prendiamo come esempio il comando DIR che abbiamo appena visto. Potreste decidere di far girare in multitasking questo comando con RUN; a questo modo, mentre viene stampata la directory, potete continuare ad usare il CLI. Esaminiamo questo comando prima della ridirezione: RUN DIR DF1:. RUN genera un messaggio del tipo [CLI 2]. DIR visualizza la directory. Quale di questi due output desideriamo ridirigere? L'output di DIR, naturalmente. Così digitiamo RUN DIR >PRT: DF1: e il nostro elenco della directory in multitasking potrà cominciare. Se comandassimo erroneamente RUN >PRT: DIR DF1:, invieremmo [CLI 2] verso la stampante e l'elenco

della directory andrebbe verso il nulla (il nuovo task creato dal comando RUN non ha una finestra di default).
Potete usare ridirezioni multiple con comandi multipli. Supponiamo che non vi piaccia il messaggio [CLI 2] che RUN genera e che non vogliate vederlo. Consegnamolo al nulla, al device NIL:.
RUN >NIL: DIR >PRT: DF1: ordina al computer di creare un processo CLI aggiuntivo (RUN), di gettare via il messaggio [CLI] (>NIL:), e di far eseguire a questo nuovo processo un comando di elenco delle directory (DIR), che invii il risultato dell'elenco verso la stampante (>PRT:).
La ridirezione dell'input è parte del sistema, ma non è molto utile con i comandi AmigaDOS. Pochi comandi usano lo schermo CLI come input di default.
Il solo comando che lo fa, di cui io sappia, sarebbe DIR quando viene usata l'opzione I (Interattivo). Se facciamo seguire il nome del comando con uno spazio e un punto di domanda, il sistema normalmente stamperà un prompt (chiamato template) e poi domanderà dell'input dallo schermo. In tal caso, possiamo ridirigere questo input. Sembra poco pratico, ma ci permetterà di mostrare la funzione.
Vediamo come funziona la ridirezione dell'input. Per prima cosa, creiamo un file di argomenti digitando ECHO >RAM: ARGS "DF0:S QUICK". Questo genera un file di una linea, usando la ridirezione dell'output; lo useremo tra un attimo. Questo file contiene la nostra risposta al messaggio di prompt che riceveremo. Digitiamo ora LIST <RAM:ARGS ?
Il sistema vede il punto di domanda e stampa come prompt il template di LIST. Ora vuole ricevere dell'input; dal momento che abbiamo ridiretto il flusso di input verso il file ram:Args, sarà là che il computer cercherà la risposta.
Osserviamo il funzionamento di questa curiosa ridirezione dell'input. Il sistema stampa il prompt di LIST e poi prende il suo input dal file RAM:ARGS. Il risultato è lo stesso che se noi avessimo digitato LIST DF0:S QUICK.

Template

Digitate un comando CLI seguito da un punto di domanda e nella maggior parte dei casi verrà stampato un prompt che contiene un template, il quale mostra le informazioni che il comando è in grado di accettare. Questi template, all'inizio, sembrano misteriosi. Una volta che li abbiate compresi, tuttavia, vi saranno molto utili. La parte interessante è che il template è usato internamente dal comando stesso. Non è solo una stringa di informazioni destinate alla visualizzazione.
Le keyword (parole chiave) entro un template possono avere speciali indicatori per specificare come devono essere usate. Ci sono tre indicatori, e io li ricordo così: /A significa Always (sempre). Dovete sempre fornire questo tipo di informazioni. /K significa Keyword. Se volete inserire questa informazione, dovete digitare la keyword. /S significa Switch (interruttore). L'uso di questa keyword abilita o disabilita un'opzione. Per esempio, digitare PROTECT ? su un sistema 1.3, genera il template FILE/A,FLAGS,ADD/S,SUB/S. Questo significa che dovete sempre fornire un nome di file, che potete specificare dei flag e che potete digitare le keyword ADD o SUB, le quali abiliteranno un interruttore del comando. Il template vi dice anche che potete fornire solo un nome di file. Usando le keyword, potete cambiare l'ordine con cui i dati vengono specificati. Non lo raccomando, perché spesso porta a degli errori, ma potreste, per esempio, copiare il file da ALFA a BETA digitando COPY TO BETA FROM ALFA invece del più usuale COPY ALFA BETA. Più utili sono i template che vengono usati internamente dai comandi e che vi permettono di porre le keyword nel luogo più conveniente. Potete dire LIST DF0: QUICK o LIST QUICK DF0:, il sistema capirà che cosa intendete.
Il comando JOIN ha un template inusuale che inizia con 15 virgole; si tratta di spazi vuoti che vi permettono di introdurre fino a 15 nomi di file sorgente. L'ultimo campo appare come AS=TO/K, che interpretiamo come "la keyword AS, oppure TO, deve essere usata per identificare questo campo". La keyword obbliga il comando a saltare ogni campo di file sorgente non utilizzato, così JOIN GIN VERMOUTH AS MARTINI usa due file di input invece dei 15 che sono consentiti.

I file di comandi

Quando avete bisogno di usare molte volte una serie di comandi, converrà scrivere un file di comandi (script) che contenga questi comandi. Digitate EXECUTE seguito dal nome del vostro file di comandi e i comandi in esso contenuti verranno eseguiti.
Sul sistema 1.3, non è necessario usare il comando EXECUTE. Una volta che abbiate abilitato il bit di script, quando digiterete il nome del file, lancerete automaticamente un'operazione di esecuzione.
I file di comandi più semplici contengono comandi che non variano mai, come la serie di istruzioni MAKEDIR RAM:WORK e CD RAM:WORK.
I file di comandi possono essere resi più creativi con l'uso degli argomenti. Un semplice esempio ve lo illustrerà. Supponiamo di avere tre file di testo, Huey, Dewey e Louie, e di volerli rinominare Hueytext, Deweytext e Louietext.
Creiamo uno file di comandi che svolga tale compito.

Usando il nostro text editor preferito, creiamo un file chiamato AddText che contenga le tre linee seguenti:

```
.key nomefile/a
RENAME <nomefile> <nomefile>Text
ECHO "<nomefile>Text RINOMINATO."
```

La prima linea dice: "Ci si riferirà alla stringa che segue AddText con l'espressione nomefile". /a significa che il nome del file deve esistere altrimenti il file di comandi rifiuterà di continuare. La seconda linea esegue il comando RENAME.
Dovunque appaia <nomefile>, verrà sostituita la stringa fornita dall'utente.
Si noti che compare uno spazio prima e dopo il primo <nomefile>, ma che il suffisso Text non è preceduto da uno spazio. La terza linea conferma che il compito è stato svolto.

Se la versione del sistema è la 1.3, abilitiamo il flag di script digitando PROTECT ADDTEXT +S. Ora possiamo convertire i tre file digitando ADDTEXT HUEY, ADDTEXT DEWEY e ADDTEXT LOUIE. Se non possedete il Workbench 1.3, digitate la parola EXECUTE prima di ciascun AddText.
Tentiamo un esempio più complesso. Supponete di avere in df1: un disco con alcune font interessanti e di volerne copiare alcune sul vostro disco di sistema. Per trasferire una font chiamata CRAZY, avreste bisogno di impartire i seguenti comandi:

```
COPY DF1:CRAZY.font FONT:
MAKEDIR FONT:CRAZY
COPY DF1:CRAZY FONT:CRAZY
```

Si noti che il nome della font deve essere digitato correttamente quattro volte. Se avete più font da spostare, potreste trovare di grande aiuto il seguente file di comandi, chiamato FontMove:

```
.key nomefont/a
COPY DF1:<nomefont>.font FONT:
MAKEDIR FONT:<nomefont>
COPY DF1:<nomefont> FONT:<nomefont>
ECHO "<nomefont> Trasferita"
```

Una volta che il vostro file di comandi sia al suo posto, potete muovere facilmente molte font, scrivendo linee di comandi come FONTMOVE CRAZY, FONTMOVE WILD o FONTMOVE RADICAL. ▲

Bit.Movie '90

La terza edizione del concorso riccionese ha avuto un ampio successo di pubblico. Due animazione realizzate con Amiga si sono piazzate al primo e al secondo posto

Giorgio Gorla

Volevate un concorso per amatori della computer animation ?
Et voilà, eccovi serviti! Nella cornice casareccia della Riccione fuori stagione, si è tenuta la terza edizione di Bit.Movie '90, mostra-concorso nazionale di computer art, dal 14 al 16 di Aprile.
L'iniziativa, patrocinata dal comune della cittadina romagnola, si è svolta presso il Centro della Pesa, "laboratorio culturale" sede della biblioteca, della videoteca, del Museo del Territorio, della sezione naturalistico-ecologica e di una speciale sezione di videoteatro e videodanza, nata sull'onda dell'annuale Premio Riccione TTVV.
Allo stesso modo, dalla ripetuta esperienza di Bit.Movie, è nata anche l'idea che il Centro della Pesa ospitasse un centro di documentazione nazionale sulla computer grafica. L'organizzazione della manifestazione riccionese è stata curata dal Circolo Culturale Ratataplan, un'associazione di liberi cittadini innamorati pazzamente dell'immagine sintetica e quasi tutti Amiga-dipendenti.

Un po' di storia

L'intenzione del Circolo Ratataplan era quella di dare spazio agli appassionati dell'immagine di sintesi fatta in casa proponendo loro di confrontarsi direttamente per misurare le proprie capacità.
Il primo concorso richiamò solo una dozzina di partecipanti provenienti da Riccione e dalla vicina Rimini, ma la presenza di un pubblico molto curioso e interessato fece sì che l'iniziativa avesse un seguito. La successiva riedizione confermò la validità dell'iniziativa: i partecipanti raddoppiarono e vennero raggiunte le 1.000-1.500 presenze.
Quest'anno si può dire sia stato un vero trionfo. I contendenti sono stati più di cinquanta e il successo di pubblico ha superato le migliori aspettative: infatti si è sfiorata la quota delle 3.000 presenze nel cor-

Waterchess

Waves

Balls

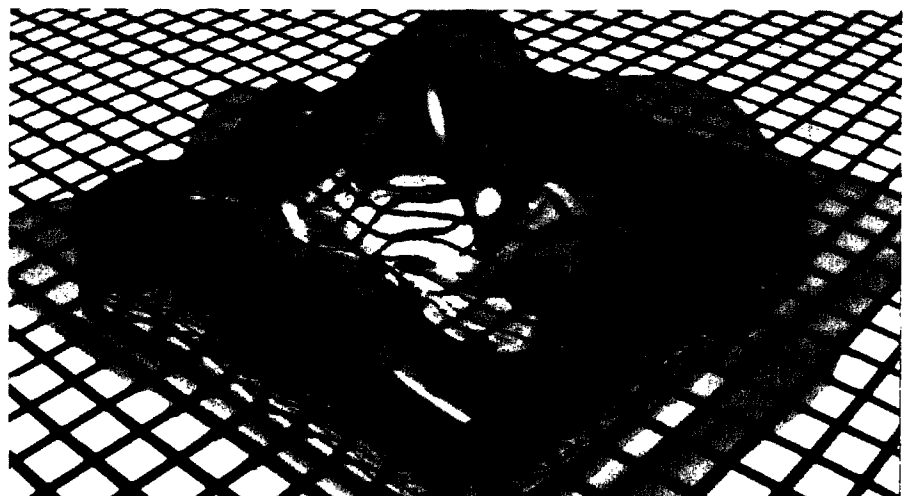
so dei tre giorni della manifestazione.
Un apposito comitato si è premurato di selezionare dalle oltre cento opere pervenute (ogni autore poteva presentare fino a quattro lavori) le trenta finaliste che hanno poi goduto degli onori della ribalta.
Al pubblico e a una giuria di esperti, composta da Bruno Magari del Circolo Ratataplan, dai giornalisti Maurizio Felletto di EnigmA, Giovanni Mazzanti, Renzo Zonin di List e dal sottoscritto, è toccato "il compito ingrato" di assegnare i tre premi in denaro: un milione di lire al primo classificato, seicentomila al secondo e quattrocentomila lire al terzo.
Come da regolamento, le animazioni ammesse alla competizione dovevano essere state realizzate con computer della classe personal e pervenire su dischetto.

Amiga protagonista

Scorrendo gli annali, esigui per la verità, di Bit.Movie, scopriamo che Amiga gioca un ruolo di primo piano. Nel corso delle passate edizioni, i partecipanti al concorso avevano presentato animazioni elaborate esclusivamente sulla bella macchina della Commodore.
Solo quest'anno hanno fatto la loro comparsa opere realizzate su piattaforme PC.
Il livello dei lavori attestava una buona "manualità" sulla macchina e, soprattutto, la validità della soluzione Commodore per lavori di grafica e di animazione a basso costo.
A giudicare dalle schede di presentazione dei lavori, il modello di Amiga preferito è risultato essere il 2000 con qualche Megabyte in più e, portafoglio permettendo, completo di scheda acceleratrice. Il software scelto dai partecipanti si può dire, a parte qualche rara eccezione, rigorosamente standard.

La competizione

Vista l'omogeneità degli strumenti, noi della giuria ci aspettavamo di avere meno motivi di imbarazzo nell'assegnare i premi in palio. Ma non è andata così. Infatti, anche il contenuto tecnico è parso ugualmente buono in tutte le animazioni passate in finale.

Le grosse differenze consistevano nei contenuti creativi e grafici. Quasi tutte le opere erano farcite di quegli elementi che nel loro insieme fanno il "kitsch" dell'immagine di sintesi: le griglie si sono sprecate; i soggetti di animazioni famose (Luxo e Wally di Lasseter sono i più gettonati) ricompaiano puntualmente; palle dalle dimensioni da incubo incombono sulle scene.
D'altra parte, il fatto che agli autori era stato fatto divieto di presentare opere su nastro, ha ridotto drasticamente il tempo tecnico a disposizione per raccontare qualcosa che avesse un capo e una coda. La ragione per cui gli organizzatori hanno scelto questa strada è quella di non favorire i partecipanti dotati di sistemi più completi e costosi.
Non sono mancate opere esclusivamente orientate alla tecnica; per esempio, uno dei lavori selezionati consisteva in un programma per Amiga scritto in Assembler con cui si possono ottenere manipolazioni dell'immagine tipiche dei processi di post-produzione televisiva (Superzoom di Marco Cozzini, Milano).

I vincitori

Per quest'anno è andata così. Al primo posto si è piazzata l'animazione "Waterchess" di Milko Mrsek, Vobarno (Brescia); la camera ruota intorno a una scacchiera dalla cui superficie liquida si innalzano e si abbattono ciclicamente i pezzi del gioco composti di ghiaccio cristallino. L'opera, tecnicamente ineccepibile, è accompagnata da una suggestiva colonna sonora. Seconda è arrivata l'animazione "The Sentinel" del californiano Bradley W. Schenck, già prima al Badge Killer Demo Contest dello scorso anno: in settanta fotogrammi è raccontata la storia di un fantasma che trova un'armatura e la indossa, ma spaventato dai tuoni e dall'atmosfera cupa del castello fugge a gambe levate.
Un piccolo aneddoto disegnato in stile fantasy, come i migliori videogiochi, con qualche buco narrativo ma una ricchezza di particolari impressionante.
Terzo, imposto dal voto della giuria e ignorato dal pubblico, una della rare animazioni realizzate con un PC IBM e grafica VGA: "Showcase" di Alberto Corti, Milano, è una simpatica animazione che vede protagoniste una serie di lampade Luxo che visionano immagini provenienti dal mondo "analogico".

La parte espositiva

Per quanto stimolante, non si può dire che il concorso sia stato il momento più importante della manifestazione. Infatti, la parte espositiva ha riservato piacevoli sorprese come la proiezione a ciclo continuo di molti lavori presentati ad Imagina dello scorso anno e ad animazioni più recenti provenienti direttamente dagli autori. Così si è potuto rivedere Locomotion, Particle Dreams, Mental Images, i cartoons 2D di Croyer, ecc. In alcune postazioni erano dimostrati il cubo nero di Steve Jobs, una stazione Iris della Silicon Graphics, le capacità editoriali di Amiga, le potenzialità didattiche di Macintosh, esempi di computer grafica 3D professionale.
Le immagini di una mostra fotografica hanno portato i visitatori nel mondo (in)credibile dell'immagine di sintesi e dell'universo frattale. Insomma, Bit.Movie ha voluto proporsi anche come importante momento divulgativo e c'è riuscita, malgrado le difficoltà di ordine pecuniario.
Infatti, gli organizzatori disponevano di un budget ridicolo; circa una ventina di milioni provenienti dal comune e dalla Graphic Delta di Rimini, unico sponsor pagante. Altre società, tra cui la Commodore Italiana, hanno sostenuto la manifestazione mettendo a disposizione materiali e servizi.
Speriamo che l'evidente successo convinca gli amministratori e i privati a perseverare e offrire contributi più adeguati. Da parte nostra auguriamo a Bit.Movie la migliore delle fortune e speriamo di rivederla più grande il prossimo anno. ▲

Scopri i segreti di AMIGA

I Servizi di

Amiga Magazine offre una serie di servizi per agevolare i propri lettori nel reperimento di software e materiale utile alla programmazione. E' disponibile l'intera libreria di dischetti di pubblico dominio curata da Fred Fish. Ogni dischetto contiene numerosi programmi e utility, spesso corredati da listati sorgenti e commenti degli autori.
Per districarsi fra le centinaia di programmi disponibili nei dischi di Fred Fish, come in precedenza su Transactor per Amiga, pubblicheremo nei prossimi mesi l'elenco ragionato.
Tale elenco riporta, divisi per categoria, tutti i programmi presenti, completandoli con informazioni quali la descrizione della funzione, l'autore, il numero di versione, la disponibilità del sorgente e il disco nel quale sono contenuti. I dischetti possono essere ordinati contrassegnando i numeri desiderati, purché la quantità sia di almeno 3 dischi.

BUONO D'ORDINE

Completare il buono d'ordine (o una sua fotocopia) e spedire in busta chiusa a:
I servizi di Amiga Magazine, Via Pola 9 - 20124 Milano

Si può allegare: assegno, contanti o fotocopia della ricevuta di versamento **c/c n.11666203** intestato a **Gruppo Editoriale Jackson.** Non si effettuano spedizioni in contrassegno.

Desidero ricevere i seguenti articoli; contrassegnare con una X i numeri di Fish Disk desiderati (minimo 3)

❑ 1	❑ 12	❑ 23	❑ 34	❑ 45	❑ 56	❑ 67	❑ 78	❑ 89	❑ 100	❑ 111	❑ 122	❑ 133	❑ 144	❑ 155	❑ 167
❑ 2	❑ 13	❑ 24	❑ 35	❑ 46	❑ 57	❑ 68	❑ 79	❑ 90	❑ 101	❑ 112	❑ 123	❑ 134	❑ 145	❑ 156	❑ 168
❑ 3	❑ 14	❑ 25	❑ 36	❑ 47	❑ 58	❑ 69	❑ 80	❑ 91	❑ 102	❑ 113	❑ 124	❑ 135	❑ 146	❑ 157	❑ 169
❑ 4	❑ 15	❑ 26	❑ 37	❑ 48	❑ 59	❑ 70	❑ 81	❑ 92	❑ 103	❑ 114	❑ 125	❑ 136	❑ 147	❑ 158	❑ 170
❑ 5	❑ 16	❑ 27	❑ 38	❑ 49	❑ 60	❑ 71	❑ 82	❑ 93	❑ 104	❑ 115	❑ 126	❑ 137	❑ 148	❑ 159	❑ 171
❑ 6	❑ 17	❑ 28	❑ 39	❑ 50	❑ 61	❑ 72	❑ 83	❑ 94	❑ 105	❑ 116	❑ 127	❑ 138	❑ 149	❑ 160	❑ 172
❑ 7	❑ 18	❑ 29	❑ 40	❑ 51	❑ 62	❑ 73	❑ 84	❑ 95	❑ 106	❑ 117	❑ 128	❑ 139	❑ 150	❑ 161	
❑ 8	❑ 19	❑ 30	❑ 41	❑ 52	❑ 63	❑ 74	❑ 85	❑ 96	❑ 107	❑ 118	❑ 129	❑ 140	❑ 151	❑ 162	
❑ 9	❑ 20	❑ 31	❑ 42	❑ 53	❑ 64	❑ 75	❑ 86	❑ 97	❑ 108	❑ 119	❑ 130	❑ 141	❑ 152	❑ 163	
❑ 10	❑ 21	❑ 32	❑ 43	❑ 54	❑ 65	❑ 76	❑ 87	❑ 98	❑ 109	❑ 120	❑ 131	❑ 142	❑ 153	❑ 165	
❑ 11	❑ 22	❑ 33	❑ 44	❑ 55	❑ 66	❑ 77	❑ 88	❑ 99	❑ 110	❑ 121	❑ 132	❑ 143	❑ 154	❑ 166	

Nota: il n° 164 non è disponibile

❑ 3 Fish Disk L. 24.000 (abbonati L.18.000)

❑ L. 5.000 per ogni disco aggiuntivo riferito ad ogni singolo ordine

❑ Interfaccia MIDI L. 92.000

Tutti i prezzi sono da intendersi IVA inclusa e spese di spedizione comprese.

Firma ______________________

(se minorenne quella di un genitore) Gli ordini non firmati non verranno evasi.

Cognome ______________________

Nome ______________________

Via ______________________

Cap __________ Città ______________

Prov __________ Telefono ______________

Codice abbonamento ______________________

ABACUS alla Grande Fiera d'Aprile

A.Borghi e A.Romano

Come accade ogni anno, forse per qualche accordo segreto, la 68° Grande Fiera d'Aprile non ha mancato di farsi accompagnare dalla solita pioggia primaverile. Quest'edizione 1990 giunge a conclusione di un ciclo d'avvenimenti che hanno portato all'ufficiale annuncio della chiusura definitiva di questa manifestazione a carattere campionario, per lasciare posto alle più specializzate fiere settoriali. Questa conclusione giunge però a giovamento degli appassionati d'informatica introducendo, in anteprima, un nuovo spazio denominato ABACUS, ovverosia un padiglione di prodotti d'informatica dove era possibile acquistare quanto presentato e offerto dagli espositori ad un prezzo particolarmente conveniente definito "offerta Fiera". La mostra era introdotta dallo stand IBM, dai toni esclusivamente d'immagine rappresentativa, accanto si trovava lo stand de "Il Sole 24 ORE" e di fianco un simpatico spazio che offriva un oroscopo computerizzato ai visitatori. Olivetti ha presentato una linea di prodotti Prodest PC, mentre Italtel Telematica sfoggiava i suoi prodotti di telefonia e un interessante terminale telematico multifunzione. In questo nuovo campo

SIP ha presentato la sua linea di telefoni, dalle più svariate foggie e colori. Ambedue le aziende telematiche, inoltre, proponevano interessanti FAX dalle prestazioni di base, sino ad arrivare ad apparecchi altamente sofisticati con caratteristiche professionali. L'area di vendita di ABACUS, delimitata ed evidenziata da ingressi realizzati con transennature a forma d'arcata, ha visto la presenza di SOFT MAIL (nota società di vendita di Software per corrispondenza), SHARP con il suo ultimo portatile AT con schermo VGA e la gamma di DATA BANK. A proposito di DATA BANK, va segnalato lo stand della Cititronics di Milano nel quale era esposta un'interessante agenda elettronica denominata 61DBMEM (THE POCKET MEMORY) dalle dimensioni di circa cm 10x6x1 con apertura a libro con uno schermo di 20 colonne per 6 righe, una capacità di memoria di 32Kb espandibile a 64K con la possibilità di trasmissione dei dati ad un altro Pocket Memory o ad un PC (tramite apposita interfaccia) per via "raggi infrarossi". Oltre a questo piccolissimo oggetto Cititronics proponeva una simpatica e curiosa calcolatrice ad energia solare dalla forma di...ABACUS (il noto sistema di calcolo che risale all'antica Mesopotamia, basato sull'utilizzo delle palline). ATARI aveva messo a disposizione dei visitatori, per quest'occasione, il suo PC FOLIO (vero e proprio PC in ambiente MS-DOS dalle dimensioni ridotte di cm 20x10x2,9 con un display a cristalli liquidi di 40 colonne per 8 righe), inoltre i più giovani hanno avuto la possibilità di provare una nuova CONSOLE portatile con uno schermo LCD a colori dalle ottime quanto sorprendenti prestazioni chiamata LYNX. COMPUTER DISCOUNT di Milano ha presentato un PC XT Compatibile, equipaggiato con due drive da 3.5" da 720 K, dalle dimensioni ridotte al punto da far impallidire un PC portatile anche perché, insieme alla CPU, c'era un altrettanto ridotto monitor monocromatico.▲

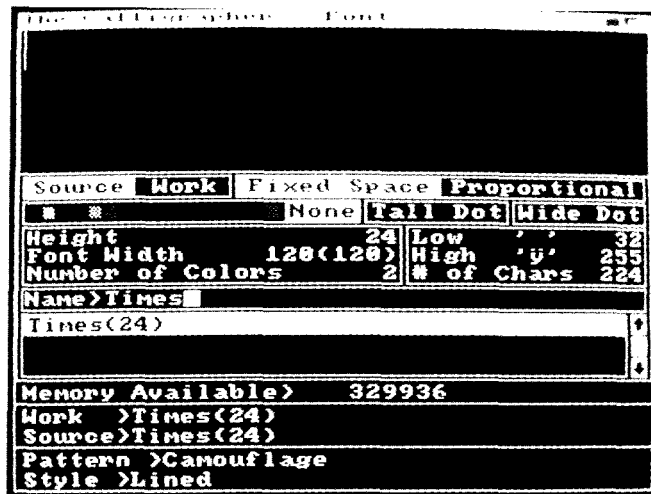
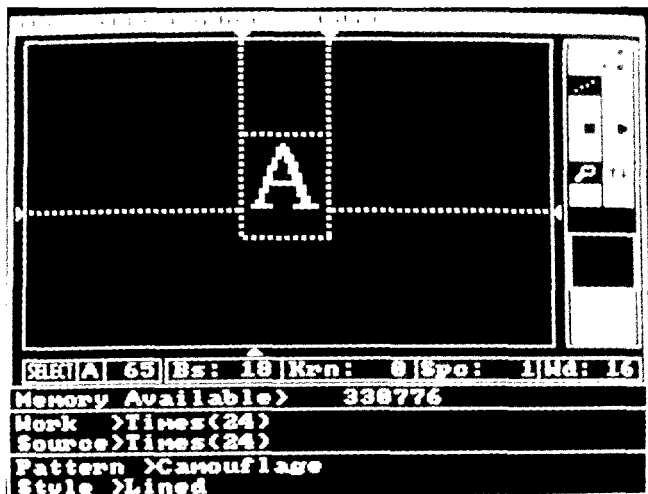
Calligrapher

Un disegnatore professionale di caratteri per Amiga.

Renato Acciardi & Daniele Cassanelli

I font sono una delle risorse indispensabili nel Desktop Publishing e nelle presentazioni videografiche; una delle caratteristiche di Amiga è quella di poter usufruire facilmente di set di caratteri diversi, prerogativa purtroppo non ancora sfruttata da una parte delle Software House, sia per problemi di spazio su disco, sia perché mancano programmi completi per la creazione di caratteri personalizzati.
Per questo motivo sono stati realizzati dei dischi contenenti unicamente font, chiamati Diskfont, che danno la possibilità di utilizzare altri caratteri, oltre ai due ROM-font residenti nel kickstart ovvero Topaz 8 e Topaz 9 che sono sempre a disposizione dei vari programmi. Un altro tipo di font sono le Colorfont, normali Diskfont con la differenza che contengono ulteriori informazioni riguardo gli attributi di colore quali sfumature, cicli o pattern.
Un font può contenere da 1 a 256 caratteri. Un Diskfont è costituito da tre elementi: il primo è un file che contiene la descrizione grafica (Bitmap) di ogni singolo carattere. Questo file ha, come nome, un numero che corrisponde all'altezza massima dei caratteri in pixel (ad esempio 8) ed è contenuto in una directory, chiamata come la famiglia di caratteri che racchiude (ad esempio Topaz), insieme ad eventuali altre altezze. Infine vi è un file dati utilizzato contemporaneamente come puntatore e come lista caratteri dai programmi che utilizzano questo font. Il nome di questo file termina con l'estensione .font (ad esempio topaz.font).
Un'altra conseguenza della mancanza di font ha portato alla realizzazione di programmi specifici per il disegno e la modifica di caratteri, che possono essere utilizzati, in seguito, da tutti i programmi che prevedono l'uso di diversi font. Uno di questi programmi è Calligrapher che possiede delle caratteristiche uniche nel suo genere.

Il Programma

Calligrapher è realizzato dalla Inter-Active Softworks, ed è utilizzabile su qualsiasi modello di Amiga con almeno 512K di memoria RAM (anche se è consigliato 1 Megabyte). Il programma si autoconfigura in versione PAL oppure NTSC. I font realizzabili, da 2 a 16 colori, sono compatibili con tutto il software disponibile che utilizza file di definizione dei caratteri in formato IFF. La minima dimensione delle font è di un pixel mentre la massima è di 256 in larghezza.
Nella confezione troviamo un disco e un manuale, abbastanza chiaro anche se poco malleabile nelle ricerche veloci.
Calligrapher, dobbiamo ammetterlo, non ha una buona presentazione: a prima vista sembra uno di quei programmi sviluppati in fretta a causa dell'incessante concorrenza che da qualche anno caratterizza il mondo del software di Amiga.
Infatti si presenta su di uno schermo 320 x 256, piuttosto confuso e disordinato. Il programma è organizzato in sei schermi diversi, attivabili dal Mode menu, che hanno diverse funzioni. Il primo è il Font Screen che viene usato per caricare, cambiare e salvare i font. Da questo screen, inoltre, è possibile vedere come i font vengono rappresentati in tutte

le risoluzioni, digitando semplicemente i caratteri da tastiera, dopo aver selezionato la risoluzione desiderata.
Il secondo schermo è il Select Screen usato per scegliere e visualizzare quali caratteri saranno usati nelle trasformazioni.
Dall'Edit Screen ogni carattere può essere modificato usando i tool standard per dipingere (che ricordano molto DPaint) come il tracciamento di linee a mano libera, ellissi, cerchi, riempimenti, ecc ...
Lo schermo Effects fornisce sei funzioni: variare l'altezza dei font da un minimo di 1 ad un massimo di 160 pixel, giustificare la spaziatura dei caratteri per l'intero set di caratteri, ridimensionare i caratteri, cambiare gli attributi (grassetto, corsivo e sottolineato) e, infine, cambiare i colori.
Da Style Screen si possono aggiungere e togliere gli stili ai caratteri, ossia è possibile aggiungere ombreggiatura, pattern, colori ai singoli caratteri.
Uno stile è formato da un massimo di sedici layer (strati o piani). Ogni layer è modificabile singolarmente (pattern e colori) ma deve mantenere la stessa forma e, importante, non deve essere confuso con i bit-plane che determinano i colori sullo schermo. Sul disco sono presenti vari stili già pronti per essere utilizzati, così come sono presenti pattern e palette già pronte.
Dallo screen Pattern si possono aggiungere pattern ai caratteri. Oltre alle classiche operazioni di caricamento e salvataggio su disco è possibile caricare dei pennelli IFF (quelli prodotti con DPaint, ad esempio) con lo scopo di creare nuovi pattern.
Il pennello non deve essere più grosso di 160 x 256 e non deve avere più di 16 colori. Una caratteristica unica di Calligrapher è la possibilità di effettuare trasformazioni: siamo in grado quindi di unire due font in uno solo, e, in generale effettuare tutte le operazioni possibili dall'Edit Screen sull'intero set di caratteri.

Le utility

Oltre al programma vero e proprio, sul disco troviamo anche alcune utility. La prima è ColorText che, attivato prima di lanciare un programma che è in grado di caricare normali font e di usare i colori, permette l'utilizzo dei caratteri a più colori. ColorText risiede in memoria, senza interferire su altro software del sistema, e intercetta le chiamate alle routine di Amiga per i font sostituendole con uno speciale programma, sempre della InterActive Softworks, che abilita i ColorFont.
I font colorati sono stati utilizzati in applicazioni come Deluxe Paint II, Deluxe Video 1.2 e Prism (con cui non è stato necessario attivare ColorText).
Un'altra utility presente è FontAssign che permette, da WorkBench, di comunicare al sistema in quale directory si possono trovare i font. Questo è importante poiché, per default, Amiga cerca i font nella directory FONTS del disco di boot, attivandolo come device logico FONTS:.
Quindi, dato che a volte è necessario caricare font da altri dischi, è necessario cambiare il path. Infatti, cliccando due volte sull'icona Font Assign da WorkBench si modifica la definizione di FONTS:. Una terza utility è FontMover ovvero un programma che permette spostamenti veloci dei font da un disco o directory ad un altro disco o ad un'altra directory.

Conclusioni

Calligrapher è un programma piuttosto completo, anche se dal punto di vista grafico lascia un po' a desiderare.
Purtroppo, come dicevamo all'inizio, come font editor ha pochi concorrenti; come editor di font a colori è ancora uno dei pochi presenti nel suo settore il quale sicuramente si svilupperà con la distribuzione della versione 2.0 del sistema operativo che, come sappiamo, supporterà pienamente i Color Font. ▲

Calligrapher è in vendita presso:
Pix Computer
Via F.D'Ovidio 6c
00137 Roma
Tel.06/8293507

a cura di A. Laus e G. Biagini

Nota dei voti

Grafica
Tiene conto della cura nel disegno dei fondali e degli sprite, dell'animazione, dello screen dei titoli.

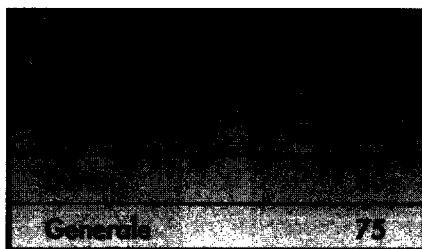
Sonoro
Valuta la colonna sonora e gli effetti sonori.

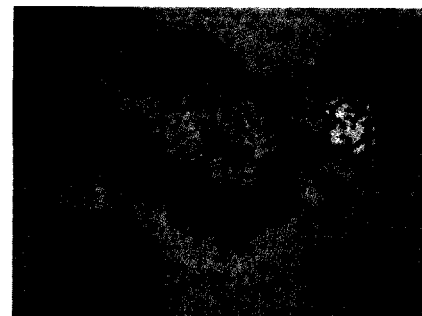
Giocabilità
Tiene conto della varietà del gioco, delle opzioni e la possibilità di giocare in due e di eventuali scelte.

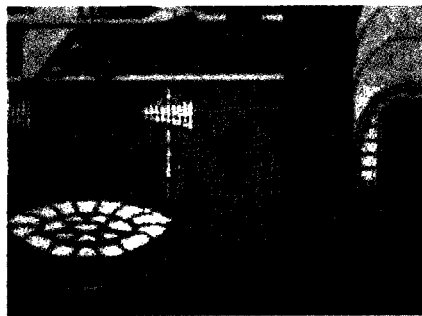
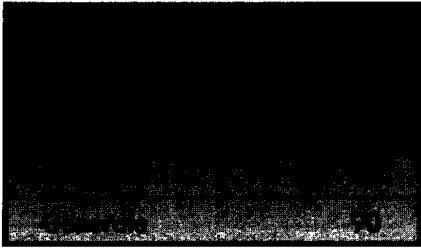
Durata
Indica per quanto tempo ci giocherete.

Generale
E' il voto generale dei valori precedenti e di tutto ciò che accompagna il gioco: confezione, manuale, gadget, ecc.

I voti vengono espressi in percentuale da 1 a 100.

Airborne Ranger

Software house: Microprose
Prezzo: 59.000 lire

La Microprose ci stupisce ancora una volta con questo gioco che ci catapulta nel duro mondo dei Ranger americani.
Il programma vi permette di vestire i panni di un Ranger Cadetto, quando volete allenarvi o giocare senza che i vostri risultati siano registrati, oppure quelli di un Veterano, per affrontare le missioni proposte dal gioco.
C'è una lunga lista di Ranger Veterani in cui potrete immedesimarvi, ciascuno preparato e adatto ad un particolare tipo di missione.
Quando vi sarete impratichiti dei comandi e del gioco nella sezione allenamento, potrete passare alle missioni vere e proprie, che, statene

certi, metteranno a dura prova i vostri riflessi e le vostre capacità di sopravvivenza.
Potete scegliere se affrontare singole missioni, ad esempio distruggere un deposito di munizioni, o se cimentarvi nella Campagna, una serie di compiti molto pericolosi che, una volta portati a termine, vi ricompenseranno della giusta gloria.
La prima missione inizia su un aereo che sorvola la zona nemica.
Voi guidate questo aereo e dovete trovare un punto abbastanza sicuro per paracadutarvi; con voi vengono lanciate, in posti strategici, le vostre dotazioni che dovrete, una volta arrivati a terra, recuperare.
Una volta a terra potete camminare, correre, strisciare, sparare, il tutto con l'uso del joystick.
Prestate molta attenzione ai campi minati che costituiscono una grave minaccia alla vostra incolumità.
La vostra dotazione di armamenti è veramente notevole e degna di un vero Ranger; l'arma che utilizzate viene evidenziata in un angolo dello schermo. Avete la possibilità di curarvi persino le ferite provocate dai proiettili nemici.
Il gioco è molto bello ed entusiasmante; curato in ogni dettaglio, vi dà l'effettiva impressione di trovarvi in una situazione reale.
L'azione è molto veloce e non vi lascia il tempo di distrarvi.
Il manuale, in italiano, vi fornisce tutte le indicazioni necessarie per impadronirvi dei comandi e delle tecniche di gioco.

Airborne Ranger è distribuito da:
Leader Distribuzione
Via Mazzini, 15 - Casciago (Va)
Tel.0332/212255

It Came from the Desert

Software house: Cinemaware
Prezzo: 49.000 lire

Un altro colpo della Cinemaware! Quando si parla di questa casa americana non può sfuggirci dalla memoria il ricordo di titoli come Defender Of The Crown, S.D.I., Sinbad oppure Rocket Ranger, tutti realizzati con una accuratezza assolutamente ineccepibile; ed ora, come si diceva, la Cinemaware è tornata alla grande sul mercato internazionale con un titolo che sicuramente otterrà numerosi consensi come i suoi emeriti predecessori: It Came From The Desert.

Questo gioco lo si può definire come una avventura grafico-testuale, ossia basata essenzialmente sulla grafica ma dove bisogna destreggiarsi anche con dei testi rappresentanti le molteplici situazioni che si possono presentare al protagonista.

Per indicare il genere dell'opera si può dire che l'avventura è ambientata a Lizard Breath, una cittadina nel deserto californiano, nella quale la vita scorre abbastanza tranquillamente senza troppi problemi.

I citati problemi però puntualmente non tardano ad arrivare quando nel deserto circostante piomba una strana meteorite e improvvisamente cominciano ad apparire delle formiche giganti che terrorizzano i tranquilli abitanti.

Sembra un incubo !

E a questo punto entriamo in scena noi che impersoniamo il Dottor Bradley, giunto da poco in città proprio per esaminare tutti i possibili indizi, rintracciati direttamente dai cittadini, che porteranno ad escogitare una strategia vincente atta ad eliminare tutte le formiche (formicaio e regina compresi...) prima che loro distruggano completamente Lizard Breath.

Come in qualsiasi avventura bisogna tenere conto di diversi fattori che possono far volgere il gioco in varie direzioni (positive o negative); ogni personaggio ha una sua caratteristica da sfruttare nel modo più opportuno e al momento giusto; ovviamente si deve sempre tenere sott'occhio il tempo (fondamentale), riposarsi quando è necessario, raccogliere le prove della natura e delle creature per guadagnarsi la collaborazione da parte dei cittadini.

Non potevano mancare certamente gli imprevisti come il faccia a faccia con le formiche (!) oppure le sfide lanciate dagli Hellcats, i bulli di Lizard Breath, nonché incendi vari e via dicendo.

Consultando il manuale allegato al gioco si possono comunque trovare dei suggerimenti riguardanti le varie situazioni che si vengono a creare in modo da uscirne nella maniera migliore.

Giudicando il gioco dal punto di vista grafico si possono dare solo pareri positivi poiché, come negli altri successi della Cinemaware, anche qui ogni scena è rappresentata eccellentemente fino nei minimi dettagli.

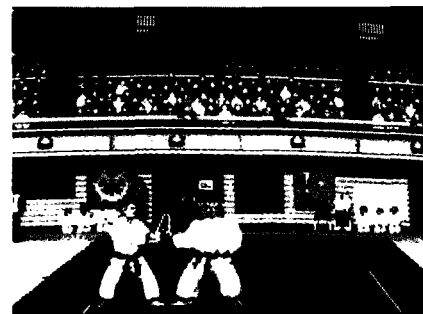
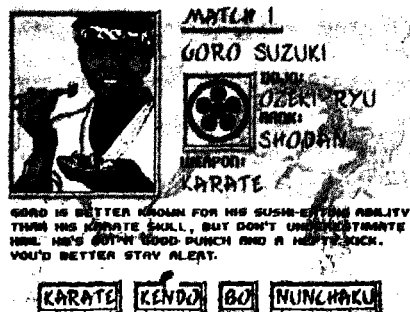
Certo, il gioco non è particolarmente consigliabile agli amanti degli shoot'em-up, ma viste le sue notevoli caratteristiche ha le carte in regola per accativarsi una buona fascia di pubblico, primi fra tutti gli appassionati di adventure.

It Came From The Desert è distribuito da: Leader Distribuzione
Via Mazzini, 15
Casciago (Va)
Tel.0332/212255

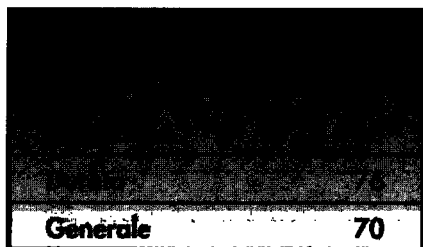
Budokan, The Martial Spirit

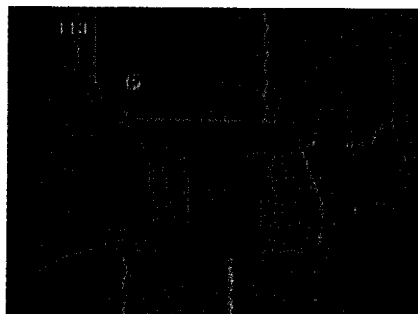
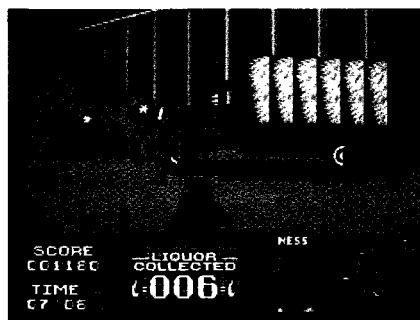
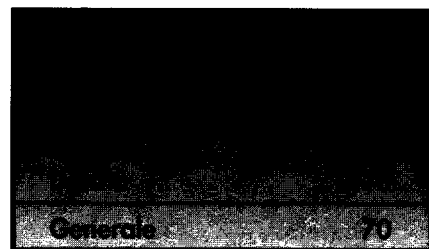
Software house: Electronic Arts
Prezzo: 59.000 lire

I maghi della Electronic Arts hanno superato veramente ogni aspettativa, dimostrando una vena creativa non comune nella realizzazione di questa eccezionale simulazione di arti marziali che racchiude in sè un realismo notevole, una grafica stupefacente, un'ottima ambientazione, degna di un film del mitico Bruce Lee, e un meraviglioso sonoro molto coinvolgente.
Degna di nota è soprattutto la facilità d'uso del vostro personaggio, caratteristica molto rara in questo tipo di simulazione, in cui si è costretti a concentrare nel Joystick decine di "mosse".

Questo programma vi avvicinerà sicuramente al mondo incantato delle arti marziali e farete fatica a smettere di giocare, ve lo assicuro.
Ma passiamo alla sua descrizione: una volta terminato il caricamento, vi trovate di fronte ai cancelli di Dojo, la scuola in cui imparerete a combattere; qui sarete messi di fronte all'unica prova spiacevole del gioco; dovrete infatti associare ad un simbolo che vi viene mostrato, una scritta (le coppie giuste le trovate sul manuale); l'unico inconveniente è che le scritte sono in giapponese.
Capisco che gli autori abbiano dovuto tutelarsi contro la pirateria, ma, almeno, le scritte le potevano fare con i caratteri occidentali!
Una volta passata questa piccola prova, a cui, in effetti, dopo un po', ci si abitua, sarete introdotti nel cortile della scuola, da cui potrete accedere ad uno dei sei edifici, ciascuno dei quali accoglie una specifica specialità.
In quattro di essi vi verrà insegnato l'uso delle quattro arti marziali previste (il Kendo, il Karate, il Bo e il Nunchaku).

Qui potrete decidere se combattere contro il vostro insegnante, di cui potrete scegliere la forza, o se allenarvi da soli.
Per i meno informati, dirò che, di queste quattro specialità, solo il Karate prevede il combattimento con l'uso del solo corpo, mentre, per le altre tre vengono usate delle armi speciali.
Nel quinto edificio, chiamato Free Spar, avrete la possibilità di combattere contro il computer o contro un altro giocatore; vi è permesso scegliere sia la specialità che userete voi che quella che utilizzerà il vostro sfidante.
Nell'ultimo edificio, infine, detto Tobiko-Sensei, andrete per sentire cosa pensa di voi il Maestro e, quando vi sentirete abbastanza forti, per iscrivervi al torneo Budokan (molto difficile da vincere).
Il gioco, contenuto in due dischetti, è accompagnato da un poster e da un eccellente manuale, in inglese, che oltre ad introdurvi nel mondo delle arti marziali, vi spiega, in modo molto dettagliato, come ottenere, con il joystick ogni singola mossa (tenete presente che ogni arte ha le sue mosse specifiche, quindi il manuale si rivela, soprattutto all'inizio, un valido aiuto per impararle).
Il giocatore 1 può usare, in alternativa, il tastierino numerico di Amiga, anche se, personalmente, consiglio l'uso del joystick.

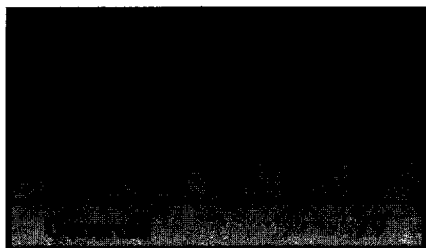
Budokan è distribuito da:
C.T.O. s.r.l.
Via Piemonte 7/F
40096 Zola Predosa (BO)
Tel.051/753133

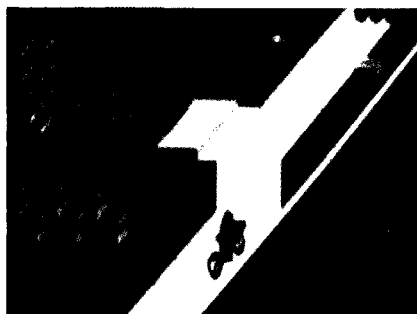
The Untouchables

Software house: Ocean
Prezzo: 29.000 lire

Come avrete certamente capito l'azione si svolge nell'epoca del proibizionismo, tra mafiosi e gangster, che combattono tra loro, a suon di mitra, per il monopolio della città.
Una cosa credo sia da mettere bene in evidenza: la OCEAN che è alla continua ricerca di un nuovo tipo di gioco, vuole rivoluzionare i suoi schemi di programmi; l'avevo già percepito quando è uscito Red Heat, e adesso ne ho la conferma. In Red Heat lo scenario era subordinato ai personaggi e rimaneva lì forse un po' trascurato visto che i soggetti erano ripresi in mezzobusto.
Qui in The Untouchables sono invece gli omini ad essere in secondo piano rispetto agli oggetti e allo sfondo: i personaggi sono molto piccoli e, a mio parere, abbastanza goffi, il proprio sprite è poco controllabile e non vi è molta interattività con il gioco.
Effettivamente l'idea poteva essere sviluppata in maniera più adeguata ma si sa, sbagliando si impara.
Da tenere comunque in considerazione la accuratissima scelta degli sfondi e la grafica curatissima.

The Untouchables è distribuito da:
Leader Distribuzione
Via Mazzini, 15 - Casciago (Va)
Tel.0332/212255

Hot Rod

Software house: Sega
Prezzo: 49.000 lire

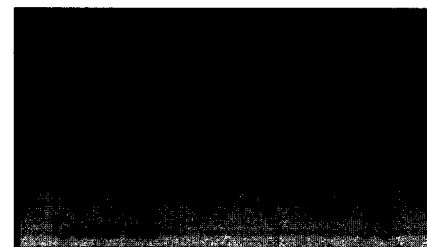
Questo gioco ricalca, in parte, quei primi arcade a sfondo automobilistico, con lo svantaggio di dovere usare il joystick e con il vantaggio di una grafica, di un sonoro e di una giocabilità più curate. Gareggiate contro altri giocatori (al massimo 3) o contro il computer allo scopo di vincere almeno 15 diverse competizioni.
Durante il percorso dovete raccogliere delle "pastiglie" di carburante, poiché la vostra vettura consuma davvero molto.
Se riuscirete a raccogliere, oltre al gas, anche dei punti Bonus, questi, alla fine di ogni gara, si trasformeranno in soldoni e vi permetteranno di comprare dei pezzi di ricambio. Il ritmo di gioco è incalzante e non potete permettervi la minima distrazione.
L'ambiente riproduce una pista di Formula 1 vista dall'alto, in cui corrono le sagome di quattro automobili, distinguibili dal colore.
E' un gioco dal tema abbastanza banale e sfruttato che, tuttavia, riuscirà ad appassionarvi e ad attaccarvi al video per un po' di tempo.
Nella confezione è presente il manuale in italiano che vi aiuterà soprattutto quando entrerete nel negozio di parti di ricambio.

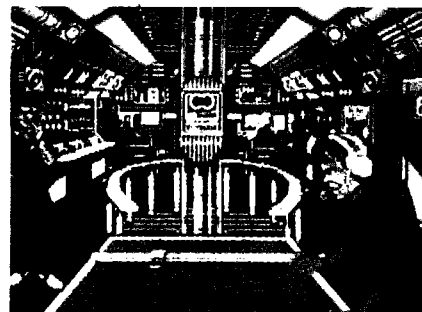
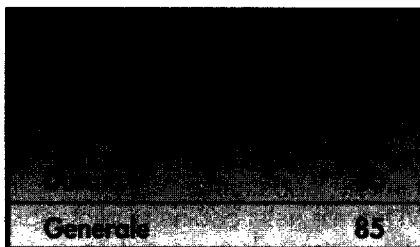
Hot Rod è distribuito da:
Leader Distribuzione
Via Mazzini, 15
Casciago (Va)
Tel.0332/212255

Paperboy

Softaware house: Elite
Prezzo: 29.000 lire

Signori, siamo di fronte ad un celebre, classico videogame per personal computer: Paperboy! Quando fu messa in circolazione la versione di questo gioco per Commodore 64, riscontrò subito i favori del pubblico poiché era considerata una delle più convincenti "traduzioni "di coin op uscite in quel periodo (penso che quasi tutti i "sessantaquattristi" del tempo ne avessero una copia). Ebbene, dopo qualche tempo, la stessa casa ha riportato il medesimo ambiente direttamente su Amiga con un risultato verosimilmente straordinario.
Il gioco consiste, come suggerisce il titolo, nel guidare un ragazzino "armato" di bicicletta (e di molta pazienza) nella consegna dei giornali direttamente a casa degli abbonati, lanciandoli tra mille pericoli e ostacoli nelle apposite cassette postali durante i giorni della settimana. I comandi quindi sono molto semplici; a complicarci la vita sono invece diversi ostacoli che assumono varie forme: si va dal pedone distratto al tagliaerba impazzito, dall'automobilista pirata al tombino scoperto, e così via.
Il ritmo del gioco è incalzante e tutti i particolari sono stati rappresentati con estrema cura e realismo, in modo tale da farci ricordare molto da vicino il tanto decantato videogioco del bar (si ringrazia Amiga per le sue potenzialità grafiche e d'animazione).
Sicuramente per la buona riuscita

del gioco bisogna imparare a coordinarsi nel momento in cui si lanciano i giornali, pena la rottura di un vetro della finestra di qualche abbonato o il travolgimento da parte di qualche sbadato passante. Divertente è la possibilità di ottenere punti rompendo i vetri delle abitazioni di chi al quotidiano non ha sottoscritto l'abbonamento (la prossima volta imparerà...).
Paperboy non presenta particolari note negative, per cui lo si può certamente promuovere a pieni voti come uno dei più divertenti e dinamici videogiochi presentati negli ultimi tempi: è un pezzo che sicuramente non mancherà nella personale videoteca degli appassionati.

Paperboy è distribuito da:
C.T.O. s.r.l.

Scramble Spirits

Software house: Sega
Prezzo: 39.000 lire

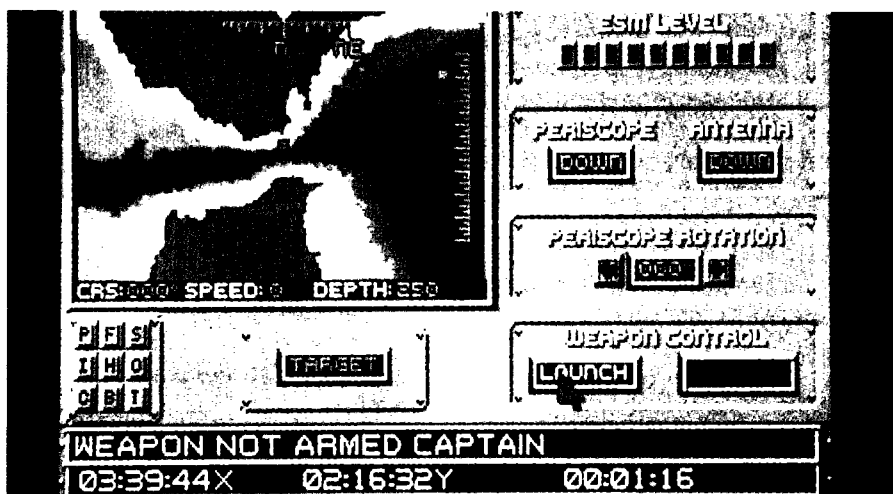
Un bel giochino dalla Sega nel classico stile Arcade Spara & Fuggi.
Nessuna sorpresa, lo scenario è abbastanza originale per quello che riguarda la grafica e il sonoro che risulta ben curato.
Il gioco comunque ricalca i soliti schemi: dovete uccidere più gente che potete, stando attenti a riportare a casa la pelle. Scramble Spirits è un gioco per uno o due giocatori, in cui voi siete al comando di un piccolo aereoplano e dovete sparare e

bombardare tutto ciò che vi circonda per la durata di sei missioni, con lo scopo ultimo di distruggere la nave di comando nemica. All'inizio del gioco avete cinque punti di credito e ciascuno di essi vi permette, una volta finita una partita di ricominciare da dove vi eravate fermati; considerando che ogni partita ha cinque vite e che avete cinque punti di credito, è molto facile terminare, anche dalle prime partite, buona parte delle missioni.
Potete volare persino in formazione, aumentando così la vostra già notevole capacità di volo. Nonostante il tema sia già stato ampiamente sfruttato, questo gioco, dalla grafica accattivante, riuscirà certamente a distrarvi per un po' di tempo dalle vostre occupazioni.
Il manuale allegato è in italiano che, se non altro, vi spiega lo scopo del gioco, visto che, in effetti, non c'è niente altro da spiegare !

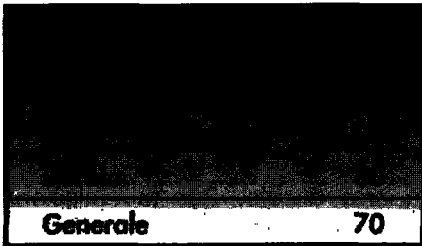
Scramble Spirits è distribuito da: Leader Distribuzione

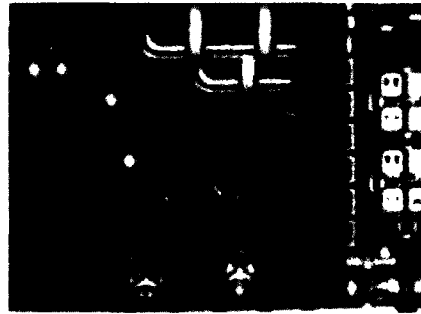
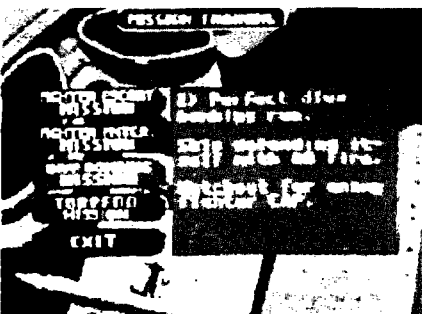
688 Attack Sub

Software house: Electronic Arts
Prezzo: 59.000 lire

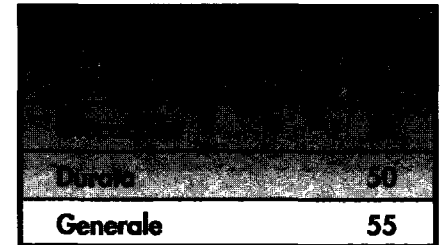
Quelli della Electronic Arts hanno fatto centro un'altra volta, sfornando un gioco davvero speciale; vi assicuro che se comincerete ad usarlo non potrete più distaccarvene. Vi è stato affidato il comando di un potentissimo sottomarino nucleare approntato per viaggiare e per combattere nelle profondità dell'oceano.Comandate, a scelta, un sottomarino americano oppure uno russo, lungo una serie di pericolose e sempre diverse missioni che richiederanno tutta la vostra determinazione. In tempo di pace, il vostro compito è quello di cercare, trovare ed eliminare le navi nemiche che si avventurano nella vostra zona di difesa; in caso di guerra, invece,

dovrete coprire le spalle alle navi da guerra alleate. Per tali missioni, sono state messe a vostra disposizione le esatte riproduzioni di tutti i più potenti sistemi di cui oggi sono equipaggiati i sottomarini come il vostro (sonar, armi nucleari e non, radio, periscopi...); potete controllare fino a 40 funzioni diverse !
La vostra missione comincia quando il Quartier Generale vi comunica il vostro compito; qui voi dovrete rispondere segnalando il vostro codice segreto personale, con cui sarete abilitati a procedere. Ci sono 10 differenti missioni che dovrete completare, la cui difficoltà è variabile e che vi porteranno in giro per il mondo; si comincia in Europa, nel Mediterraneo, si passa al Baltico, all'Atlantico, fino ad arrivare nel Pacifico.
In ogni missione vi è richiesto di preparare un piano di attacco e di studiare la zona, preliminari fondamentali per portare a casa la pelle.
Ci sono sia missioni di difesa che missioni di attacco, ciascuna spiegata nei dettagli nel manuale, in inglese. Il manuale vi spiega anche i vari comandi del vostro sottomarino e riporta un elenco delle maggiori navi da guerra e delle loro caratteristiche fondamentali. La cosa che più colpisce è certamente la grafica, avvincente, che rappresenta un ulteriore stimolo a diventare dei bravi capitani. Il sonoro è ben fatto e rende bene le situazioni; è possibile giocare con il mouse, ve lo consiglio, o con la tastiera. Sebbene non sia uno di quei giochi in cui si diventa subito bravi, perché richiedono poco impegno, 688 Attack Sub vi appassionerà fin dall'inizio.

Attack Sub è distribuito da: C.T.O.

Battlehawks 1942

Software house: Lucasfilm Games
Prezzo: 29.000 lire

Si tratta di un buon simulatore di volo, inserito nell'ambiente delle battaglie aereo-navali della II° guerra mondiale.
Con questo gioco, spiega il manuale, rivivrete la storia e, se sarete abbastanza in gamba, potrete persino riscriverla.
In effetti si tratta di una buona simulazione della realtà, in particolare per la particolare drammaticità dell'azione; non posso, purtroppo dire altrettanto per i comandi che risultano, rispetto alla velocità degli avvenimenti, abbastanza lenti.

Vi trovate nella cabina di pilotaggio del vostro velivolo e potete avere una visuale frontale, laterale e posteriore, sebbene il cambiamento di vista vi faccia perdere qualche secondo, a volte vitale, di gioco.
Il gioco vi permette di scegliere il tipo di aereo da usare, la missione da compiere e persino il luogo dello scontro.
Potete scegliere l'addestramento o le missioni vere e proprie; comunque i vostri risultati verranno segnati sul vostro libretto di volo personale.
Potete ispezionare, prima di partire, tutti gli aerei, valutandone così i punti di forza e i punti deboli.
Le missioni sono sempre ambientate sul mare; dopo ogni missione verrete valutati e i vostri risultati saranno registrati.
E' possibile persino utilizzare una telecamera che filma le gesta del vostro aereo, per poi mostrarvele, come se vedeste un film.
Il manuale, in inglese, è diviso in più sezioni: si passa dalla storia delle battaglie aereo-navali, alla spiegazione del gioco, dai primi rudimenti del volo, alle tecniche di combattimento, alla descrizione degli aerei.
Viene allegato anche un fascicolo, questa volta in italiano, in cui sono riassunti i comandi di volo e le varie missioni.
Si tratta, nel complesso, di una buona simulazione, anche se presenta qualche carenza nella velocità di gioco.

Battlehawks è distribuito:
Leader Distribuzione
Via Mazzini, 15 - Casciago (Va)
Tel.0332/212255

Outlands

Software house: Pandora
Prezzo: 29.000 lire

Ci risiamo: hanno preso Amiga per una macchina a 8 bit.
Ringraziamo sentitamente le Software House come la Pandora che ci propongono degli spaziali stile Commodore 64 per non farci morire di nostalgia, però trattiamo Amiga per quello che vale!
Cari amici della Pandora, non metto in dubbio che il prodotto sia stato curato molto sotto il profilo grafico (il risultato non è comunque eccezionale), però gli scroll a scatti e la poca giocabilità del vostro programma non corrispondono affatto alla descrizione sul retro della confezione, si parla di quattro massicci livelli con scroll fluidi e i più raccapriccianti mostri mai visti: non credo sia completamente vero questo, poiché il programma in questione altro non è che uno Shoot'em-Up abbastanza lento con tecniche di scroll particolarmente sorpassate e il continuo passaggio di file di nemici (molto simili fra loro) che cercano di uccidervi.
Molto più interessante è il gioco a due: si può infatti combattere simultaneamente (la cosa più divertente è rubarsi i Bonus a vicenda). In conclusione bella grafica ma gioco poco appassionante (perché troppo ripetitivo).

Outlands è distribuito da:
Softel
Via E.Dal Pozzo, 7
00146 Roma
Tel.06/5584334

E' IN EDICOLA

GRANDE AMIGA

LA PRIMA GRANDE RACCOLTA DI:

3 RIVISTE E 2 FLOPPY SOLO A L.10.000

TUTTO
PROPRIO TUTTO
PER GIOCARE
PROGRAMMARE
E CONOSCERE
SEMPRE DI PIU' IL TUO AMIGA

**RICHIEDI "GRANDE AMIGA"
NELLA TUA EDICOLA DI FIDUCIA**